AN INTERACTIVE INFORMATION RETRIEVAL SYSTEM SORGU

by

NAMIK KURAL

B.Sc. in Computer Engineering

Middle East Technical University, 1981

Submitted to the Institute for Graduate Studies in Science and Engineering in partial fullfillment of the requirements for the degree of

Master of Science

in

Computer Engineering

Boğaziçi University

1984

AN INTERACTIVE INFORMATION
RETRIEVAL SYSTEM - SORGU

APPROVED BY

Doç. Dr. Tunç Balman .....Balman.....................

Doç. Dr. Gündüz Ulusoy ....

Doç. Dr. Selahattin Kuru ....

DATE OF APPROVAL:

ACKNOWLEDGEMENTS

This study was conducted under the supervision of Doç. Dr. Tunç Balman to whom I wish to express my sincere thanks for his invaluable guidance, encouragement and support.

I am also grateful to Doç. Dr. Gündüz Ulusoy and Doç. Dr. Selahattin Kuru for serving on my thesis committee.

In particular, I would like to thank Mrs. Ayşegül Kural for her support and patiance.

ABSTRACT

There are several interactive information retrieval software packages that maintain easy access to data base items and structures. It is relatively easy for persons familiar with computer to use these packages. But to persons who are unfamiliar, especially the managers, using these packages is not easy. The syntax of these packages is extraordinarily different for these persons and usually do not wish to spend time learning this syntax.

SORGU is a turkish based interactive information retrieval software package. The queries are formed using menus. It does not have any syntax convention. Anybody who has knowledge about the fundamental concepts of data base management systems can easily use SORGU. It is a completely menu driven inquiry system that provides additional facilities to change the item naming conventions, supports multiple enquiries and user defined report structures. Because of the ease of use training time is minimized.

ÖZET

Veri tabanlarından bilgiye kolayca erişebilmeyi sağlayan çeşitli etkileşimli bilgi erişim yazılım paketleri hazırlanmıştır. Bilgisayara yatkın olan özellikle ingilizce bilen kişiler için bu paketleri kullanmak oldukça kolaydır. Ancak diğer kişilerin özellikle yöneticilerin bu yazılım paketlerini kullanmaları çok zordur; bu paketleri kullanmak için gerekli olan sözdimizi düzenleri bu kullanıcılar için oldukça olağan dışıdır, ayrıca öğrenim için ayrılabilecek zaman kısıtlıdır.

SORGU Türkçe'ye dayanan bir etkileşimli bilgi erişim yazılım paketidir. Sorular menular aracılığı ile oluşturulur. Belirli bir sözdizimi düzeni yoktur. Veri tabanı yönetim sistemleri temelleri hakkında bilgisi olan bir kişi rahatlıkla kullanabilir. Sistem tamamiyle menular aracılığı ile kullanıcıya şablon tanımlama, çoklu sorgulama ve basit raporlar hazırlama opsiyonlarını da ayrıca içerir. Kullanım kolaylığından dolayı öğrenim süresi asgaridir.

TABLE OF CONTENTS

LIST OF FIGURES

I. INTRODUCTION

A Management Information System which contains a data base is an essential system that an enterprise should have. The decision making and transaction processing functions in an enterprise can easily be handled after installing this system. Handling queries for obtaining information from the data base requires an on-line inquiry software.

Information retrieval is the application of data bases and data base systems to the processing of queries where the result is expected to increase the users' knowledge. Data base applications which are not considered information retrieval are operational service systems, systems which schedule activities, manage inventories, and prepare bills, etc. In order to develop a basis for comparison of information retrieval opjectives, one can categorize information retrieval into three areas:

i) Fact finding: selection and output of data from the data base

ii) Statistical inference: summarization of data subsets

iii) Deductive inference: exploration of relationships implied in the data base. (1)[x]

Query formulation is the process performed by the user in order to communicate through the information retrieval system with the data base. Many languages to state queries have been developed. They range from very formal, set-mathematics oriented languages to relatively simple languages which provide statements similar to those found in procedural languages. The user of statement-oriented languages is typically intermediary between the requestor of the information and the system. Some training is required to use the available facilities and the system. The specification of

[x] Numbers enclosed in brackets refer to the references at the end

queries with many parameters tends to be awkward when not supported by forms or formatted display screens. It seems then much more desirable to allow natural language like input.

Since the information retrieval software are general purpose they have some disadvantages. Usually, these packages resemble a natural language but this language may not be the native language of the user. Also a syhtax convention similar to programming languages has been formed in order to form the queries, but understanding the syntax may sometimes be too hard for a user unfamiliar to the computer. The user should use the item names in the data base which may be too long to be self-explanatory, or too short which does not have any meaning; therefore the user should be capable of defining new names for the item names hence defining mapping relations. Also some machine dependent problems may arise such as line at a time processing, etc.

For Burroughs large systems there exists a data base management system software called Data Management System II (DMS-II) that has an on-line inquiry facility. SORGU the turkish version of INQUIRY - the on-line information retrieval software of DMS-II, has been prepared to overcome the above disadvantages.

The first chapter deals with the general concepts of Management Information Systems, Data Base Management Systems, and DMS-II.

The third chapter shows a real-life application; queries are formed and their results are explained.

The fourth chapter deals with the technical attributes of SORGU and explains show to use SORGU.

II. DATA BASE SYSTEMS AND DMS II

A management information system, or MIS is an information system that provides all necessary transaction processing for an organization also provides information and processing support for management and decision functions. The computer has added a new and powerful technology to information systems, so that the computer-based information systems can be radically different from systems using manual or electromechanical processing. A formal definition of a management information system is an integrated, man machine system for providing information to support the operations, management, and decision making functions in an organization. (2) The man machine system concept implies that some tasks are best performed by man, while others are best done by machine. An integrated system is based on the concept that there should be integration of data and processing. Data integration is accompleshed by the data base. The advanced information processing system must still provide for processing of transactions. The trend in transaction processing in advanced systems is towards online data collection and online inquiry. The online inquiry capability is very significant in operational support. It means that any authorized employee may obtain immediate response to an inquiry such as the current balance in a customer account or the inventory on hand for an item.

2.1. Data Base Systems

Definition of a data base before going into the details is as follows - A data base is a collection of stored operational data used by the application systems of some particular enterprise. Advantages gained by using the data base should be discussed in order to explain the concept clearer. The term data base administrator should be defined as a person who has the central responsibility for the operational data. (3) The amount of data redundancy can be reduced. In most current systems each application has its own private files where most of them contain duplicated information The stored data can be shared, It means not only that all the files of existing applications are integrated, but also that new applications may be developed to operate against the existing database. Standards can be enforced. With central control of the database, the DBA can ensure that installation and industry standards are followed in the representation of data. Security restrictions can be applied. Data integrity can be maintained. Centralized control of the database helps in avoiding integrity problems by permitting the DBA to define validation procedures to be carried out whenever any storage operation is attempted.

One may categorize database systems according to the approach of handling the data model. The three best known approaches are

The relational approach

The hierarchical approach

The network approach,

The main difference between these approaches is the way in which they permit the user to view and manipulate associations.

In the relational approach associations are represented in the same manner as other entities. In the hierarchical and network approaches certain associations are represented by means of "links". Basically such links are capable of representing one-to-many associations; the difference between the network and hierarchical approaches is that with the former links may be combined to model more complex many to many associations, whereas this is not possible with the latter. (3)

2.2. DMS II

Burroughs' Data Management SYstem-II (DMS-II) is a comprehensive data base management system that interacts with the Master Control Program (MCP) operating system to give its users a viable DBMS working environment. The access routines are tailored for each defined data set structure, are loaded into main memory when the data set is invoked, and then operate as MCP operating system intrinsics.

DMS-II was initially announced by Burroughs in October 1974, and was designed to replace the earlier DM 700 data base management system that was available for the large scale Burroughs computers. Burroughs took a different approach in the design of DMS-II by integrating some of the data base management routines into the MCP operating system. The system represents Burroughs' own definition of the requirements for a sophisticated data base management system and does not follow the format of the CODASYL Data Base syntax or architecture (although the UCLA Extensior Data Base Conference rates DMS-II as a "CODASYL-like" implementation). (4)

DMS-II consists of a stand-alone Data and Structure Definition Language (DASDL) for defining the information in the data base, establishing relationships between the data records, and mapping the data base to direct access storage devices, plus extensions to the ALGOL, PL/I, and COBOL compliers to provide commands for manipulating data in the data base. Data base structure definitions are stored in a Data Base Description file. DMS-II uses the Description file to produce re-entrant access routines that are loaded into the main memory and appended to the MCP only when the data base structure is opened. DMS-II operates under the Burroughs MCP operating system in batch, transaction processing, remote job entry, and time-sharing processing environments.

2.3. Characteristics of DMS-II

The DASDL complier processes the DASDL source statements and creates a DASDL Description File on direct access storage which contains a complete description of the structural characteristics of the data base. The DASDL Description File is accessed by the host-language compliers, which automatically insert a description of the invoked portions of the data base in the program at the compile time. The Description File is also accessed by the DMS-II component that creates specialized access routines used by programs in accessing the data base.

The DASDL recognizes three basic entities within a DMS-II data base: "data sets", "sets", and "items". A data set is a logical file, or collection of related records consisting of the actual information in the data base. Data sets can be "embedded" or "disjointed". An embedded data set is an element of another data set and defines a hierarchical relationship between a Master record and Detail records in the embedded data set. A disjoint data set is a free standing data structure which can act as a root of a tree structure. Disjoint data sets can be linked to form network structures in the data base.

DMS-II also supports a global data capability for collecting summary or statistical information related to a data set or to entire data base.

Physical records in the data are logically related using "sets", "subsets", and "links". Set definitions are represented by key tables; they describe the keys for accessing the records in the data set and the retrieval methods to be used for accessing records in the data set, plus various options for organizing the data set (e.g. ordering of records, presence of duplicate keys, etc.) Each set (key table) contains a path into the data base for every record in the data base, and there can be

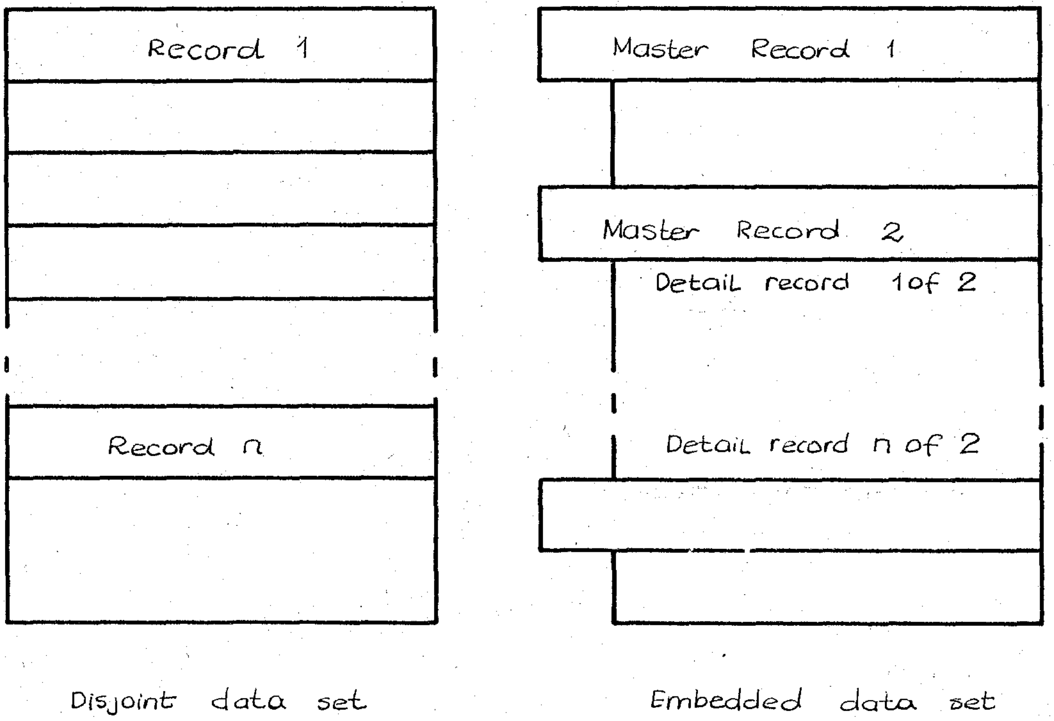


Fig. 2.1 Data Sets in DMS II

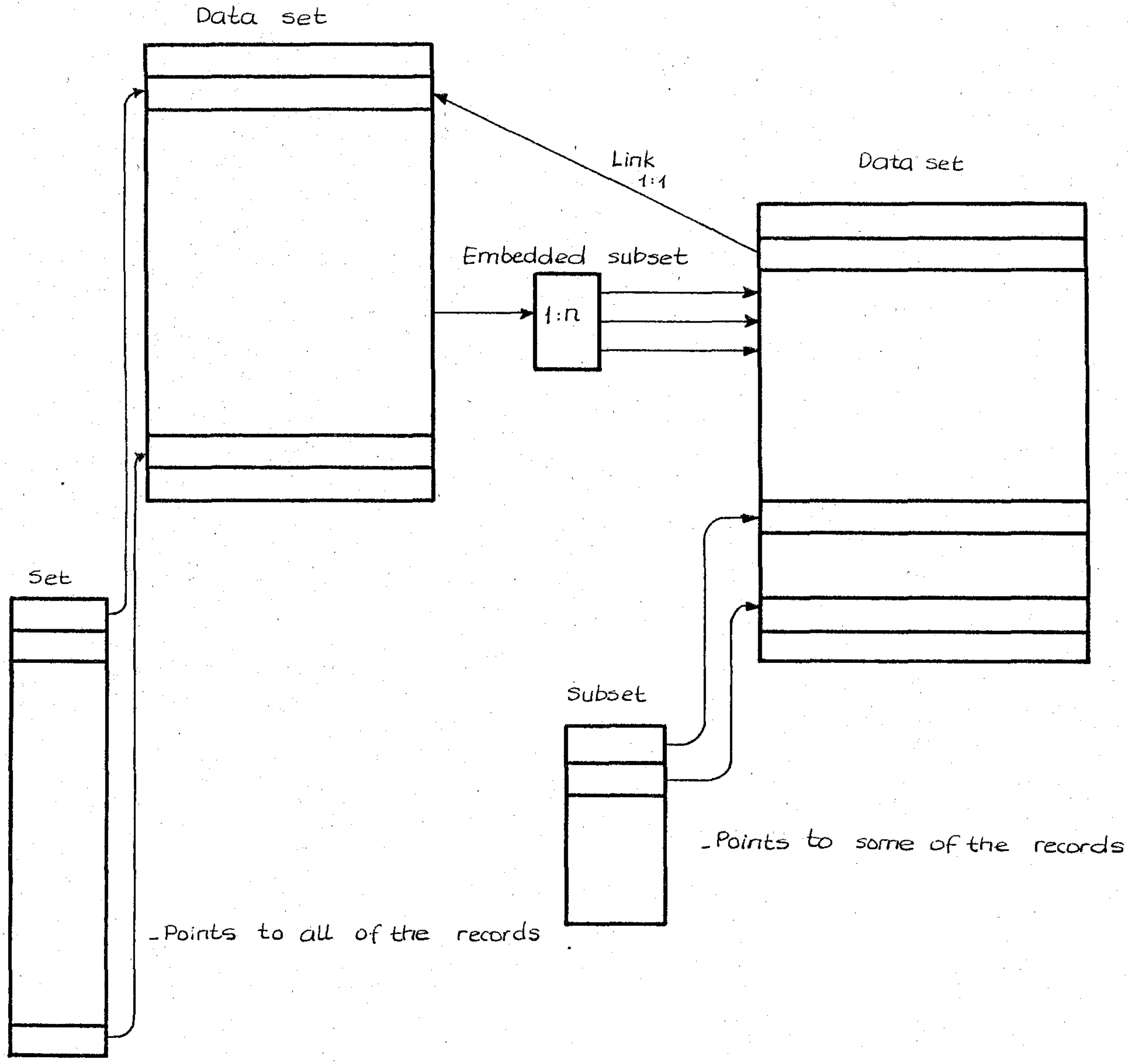


Fig. 2.2. Set, Subset and Links in DMS II

any number of key tables associated with a single data set to provide multiple entries to the records in the data base. Subsets or partial collection of information, can also be defined in the set definition, allowing selected members of a data set to be retrieved based on a specified selection criterion.

All sets in DMS-II are automatic, so that the addition or deletion of records is performed by the data base management system for all other related sets. Subsets can be automatic or manual. In manual subsets, each new entry or deletion must be performed by the programmer. In automatic subsets, each new entry to the data base is examined for conformance to subset criteria and is automatically inserted into the appropriate sets. Subsets can also be embedded within the data set, with a separate key index maintained for the defined subsets. Embedded subsets allow a member of a data set to be logically related to members of other data set on a one-to-many basis, and are maintained as automatic subsets. Link relationships can be specified between members of data sets on a one-to-one basis. When links are used, DMS-II maintain a pointer (a relative disk address) in the owner record to the related member, with information permitting direct retrieval of the data, generally in one or two disk accesses.

The DASDL provides options for controlling the mapping of the data base to direct-access storage. Data sets can be defined as standard unordered, random, direct, compact, or restart. Options are also available for specifying the block sizes, the number of areas, and the size of the areas on disk storage, and the number of buffers in memory to be assigned to each data set. Buffers are automatically maintained as a system buffer pool by the MCP. The DASDL also allows the user to specify key table sizes and to supply a load factor to control the addition of entries to the key table. A new key table is automatically created when the load factor is reached to avoid key table overflow and reorganization.

DMS-II provides host-language interfaces for the COBOL, ALGOL, and PL/1 languages. A compile time interface, called the Data base Interface, is a seperately compiled co-routine that supplies the host-language compiler with the description of the selected portion of the data base. The Database Interface also verifies that the syntax of the data base manipulation statements is valid for the invoked data base structure and generates calls on procedures in the Run-Time Interface. The Run-Time Interface is activated when the data base is opened, and maintains one or more record areas for each invoked data set and one or more paths corresponding to each set invoked by the program. A program can request multiple record areas for concurrently concurrently accessing more than one record in a data set and can issue multiple calls to a single, re-entrant access routine. A current-record pointer is associated with each record area, and a path to the most recently accessed record is associated with each set.

The data manipulation verbs operate as an extension to the host-language compiler and consist of the following commands: OPEN, CLOSE, LOCK, STORE, DELETE, FREE, FIND, SET, CREATE, RECREATE, GENERATE, INSERT, REMOVE, ASSIGN, BEGIN TRANSACTION, and END TRANSACTION.

DMS-II provides audit and recovery capabilities to automatically recover the data base from Halt/Load conditions, to provide restart information for user programs, to reconstruct portions of the data base, and to remove aborted transactions. The system maintains (if specified in the DASDL definition of the data base) an audit trail of all before and after images of modifications to the data base signified by BEGIN TRANSACTION and END TRANSACTION statements. Both syncpoints and control points are specified at regular intervals on the audit trail. Syncpoints are used for recovery from Halt/Load conditions and aborted transactions, and are recorded at the completion of a user-specified number of transactions. Control points are taken on the audit trail at again user-specified syncpoint intervals and includes dumps of the system buffers. Buffers are dumped at least once every two control points to facilitate recovery.

Revovery from a system failure is initiated automatically after a Halt/Load condition. Before images on the audit trail are scanned backward to the last syncpoint to restore the data base to its original condition, and after images from the second last control point are used to update the data base with the results of completed transactions. A Restart Data Set containing the contents of the restart record area for each program is used to recover from aborted transactions. An on-line dump and data recovery utility operates concurrently with data base processing and creates backup copies of pages in the data base for recovery from disk errors.

Data base reorganization utilities are generated in response to special DASDL syntax for such operations as adding or deleting structures or items and reclaming disk space. Reorganization of the data base does not require changing or recompiling of application programs.

DMS-II also includes a Reporter System for generating reports using a free-from vocabulary, and the INQUIRY language for browsing through the data base or retrieving records based on usersupplied selection criteria.

2.4. DMS II INQUIRY

INQUIRY provides a simple method by which a user can examine information in a DMS-II data base. The examination is usually done by using a terminal. Some characteristics of the INQUIRY are as follows:

- INQUIRY can examine information in any part of the data base.

- INQUIRY can update, create, and delete records.

- INQUIRY takes advantages of sets, if possible, in extracting information from a data base.

- INQUIRY always produces the requested information, even if it is necassary to perform linear searches to satisfy the request.

- INQUIRY allows user to examine the description of the data base.

- INQUIRY contains relatively few statements; but these can be combined to perform complex operations.

- INQUIRY can generate simple reports from data contained in DMS-II data bases. (5)

2.5. INQUIRY Statements

ATTACH Allows the user to combine an embedded structure with its owner to establish automatic looping between the two structures.

CLEAR Discards DEFINE items, VIRTUAL items, and/or GENERATEd subsets.

CREATE Causes a new record to be created in the data base.

DEFINE Allows INQUIRY text to be assigned a name.

DELETE Causes a record to be deleted from the data base.

DETACH Separates an embedded structure from its owner to prevent automatic looping between the two structures.

DISPLAY Allows items of selected record to be displayed.

EDIT Allows a previous INQUIRY statement to be modified without requiring the entire statement to be re-entered.

GENERATE Creates a temporary subset of a data set.

HELP Displays the syntax and semantics for each INQUIRY statment. It may also be used to obtain additional diagnostic information following an error.

NEXT Causes INQUIRY to resume record selection and item display.

OPTIONS Allows INQUIRY options to be displayed or altered.

PRINTER Allows the attributes of the line printer file to be displayed or altered.

QUIT Terminates the INQUIRY session.

RECALL Retrieves the text of a prior INQUIRY statement

REPEAT Causes re-execution of a previous INQUIRY statement.

RESTORE Allows previously SAVEd text to be retrieved.

SAVE Stores the text of DEFINE items, VIRTUAL items, and GENERATEd subsets in a file on disk.

SELECT Locates records which satisfy the selection criteria specified by the user.

SET Modifies or deletes the text of the most recently entered DISPLAY, REPEAT, SELECT, or UPDATE statement for a given data set.

REPORT Controls both the type of items that are to be listed on a report as well as the format of the report.

SHOW Displays all or selected portions of the data base description and may also be used to display the most recently entered INQUIRY statement.

SUMMARY Provides statistical information about items contained in a report list.

SORT Allows a user to control the amount of core and disk used by the SORT option.

TERMINAL Allows the attributes of the terminal file to be displayed or altered.

TITLE Defines title characteristics of a report.

UPDATE Modifies items in a selected record.

VIRTUAL Allows new items to be defined which are functions of other items.

2.6. On-line Information Retrieval

Although the computer offers many advantages in information retrieval activities, the off-line batch processing systems have disadvantages. They are essentially "one-change" searching systems in which the user has to think in advance of all possible search approaches and construct a search strategy that is likely to retrieve all relevant data. Another disadvantage of the off-line system is that the search results are usually delay - you can not get an immediate response. Another major disadvantage is that the user should explain his needs to a specialist to get the results, but the user may be unable to explain clearly what he is seeking or the specialist may misinterpret the real needs of the user.

The term on-line refers to the fact that the user is in direct communication ("on-line to") with the data base he wishes to access. An inquiry is conducted as a two way conversation between the user and the system (computer). For this reason the on-line system is frequently referred to as interactive or conversational. As well as being referred to as interactive, or conversational an on-line system is frequently associated with the adjectives "time-shared", and "real-time". On-line time sharing implies the sharing of machine processing time among a number of terminals. Real-time operation implies that the computer receives data, processes it, and returns quickly enough for them to be used in some ongoing job. (6)

III. REAL LIFE EXAMPLE OF SORGU

In this chapter a real life application of SORGU will be presented. Typical queries that can be formed using SORGU are shown and explained. The inputs are represented by "→" or they reside within the delimeters and "◁".

The following instruction imitiates a SORGU session named İKMAL:

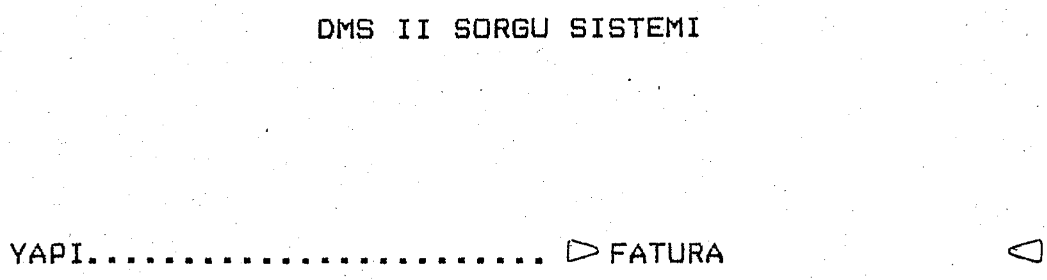
```
→ R SORGU/IKMAL
```

```
#RUNNING 5287
#?
#B6800 MIS SORGU   3.3.320.080
#SORGU BASLIYOR
```

The main menu is displayed and option 1 is elected in order to fund the records fulfilling particular set of conditions.

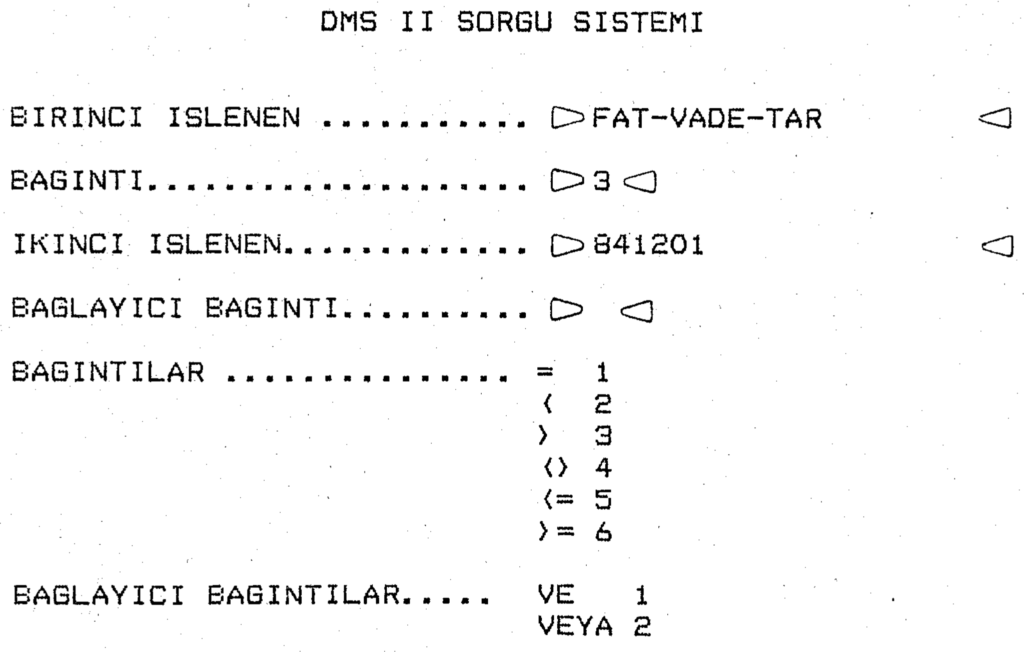
```
                        DMS II SORGU SISTEMI
        SEC................................................   1
        GOSTER.............................................   2
        AT.................................................   3
        YARAT..............................................   4
        GUNLE..............................................   5
        SONRAKI............................................   6
        BITIR..............................................   7
        GORUNTULE..........................................   8
        RAPORLA............................................   9
        TANIMLA............................................   10
        TANIMLAMA AT.......................................   11
        SAKLA..............................................   12
        YUKLE..............................................   13
        SABLON-EKLE........................................   14
        SABLON-AT..........................................   15
        SABLON-SAKLA.......................................   16
        SABLON-YUKLE.......................................   17
        TEKRARLA...........................................   18
        YAZICI.............................................   19
        Lutfen yukaridaki seceneklerden birini giriniz   ▷  ◁
```

Structure FATURA is elected



```
                 DMS II SORGU SISTEMI



YAPI......................... ▷ FATURA                 ◁
```

Search criteria is defined as FAT-VADE-TAR 841201



```
                 DMS II SORGU SISTEMI

BIRINCI ISLENEN ........... ▷FAT-VADE-TAR            ◁

BAGINTI.................... ▷3 ◁

IKINCI ISLENEN............. ▷841201                  ◁

BAGLAYICI BAGINTI.......... ▷  ◁

BAGINTILAR ...............  =   1
                            <   2
                            >   3
                            <>  4
                            <=  5
                            >=  6

BAGLAYICI BAGINTILAR.....   VE    1
                            VEYA  2
```

#

Main Menu is displayed after the system located the requested record. Option 2 is elected in order to see the contents of the locates record.

```
                   DMS II SORGU SISTEMI
SEC............................................  1
GOSTER.........................................  2
AT.............................................  3
YARAT..........................................  4
GUNLE..........................................  5
SONRAKI........................................  6
BITIR..........................................  7
GORUNTULE......................................  8
RAPORLA........................................  9
TANIMLA........................................  10
TANIMLAMA AT...................................  11
SAKLA..........................................  12
YUKLE..........................................  13
SABLON-EKLE....................................  14
SABLON-AT......................................  15
SABLON-SAKLA...................................  16
SABLON-YUKLE...................................  17
TEKRARLA.......................................  18
YAZICI.........................................  19
Lutfen yukaridaki seceneklerden birini giriniz   ▷ 2◁
```

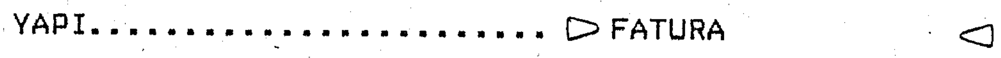
The same structure "FATURA" will be used.

```
YAPI.........................
```

The record is displayed.

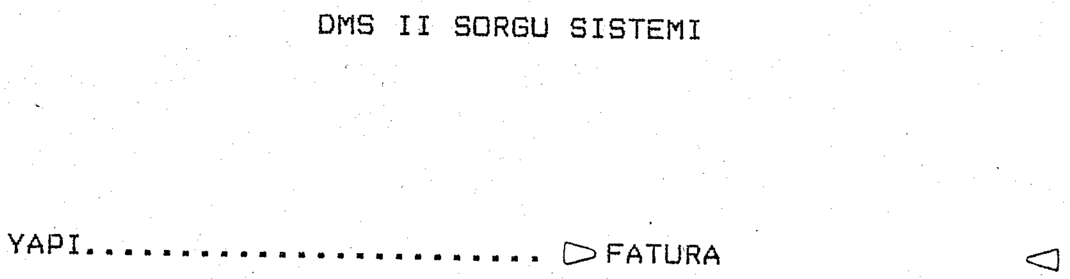
```
 FAT-NO  FAT-YUKLEME-TAR  FAT-VADE-TAR      FAT-TUTARI      FAT-BAKIYESI
-SIPNO
---------------------------------------------------------------------------
  11584          840712        850112   70000000.0000      70000000.00
 14684
```

Option to delete (3) is elected at the main menu. The user elects to delete structure "FATURA"



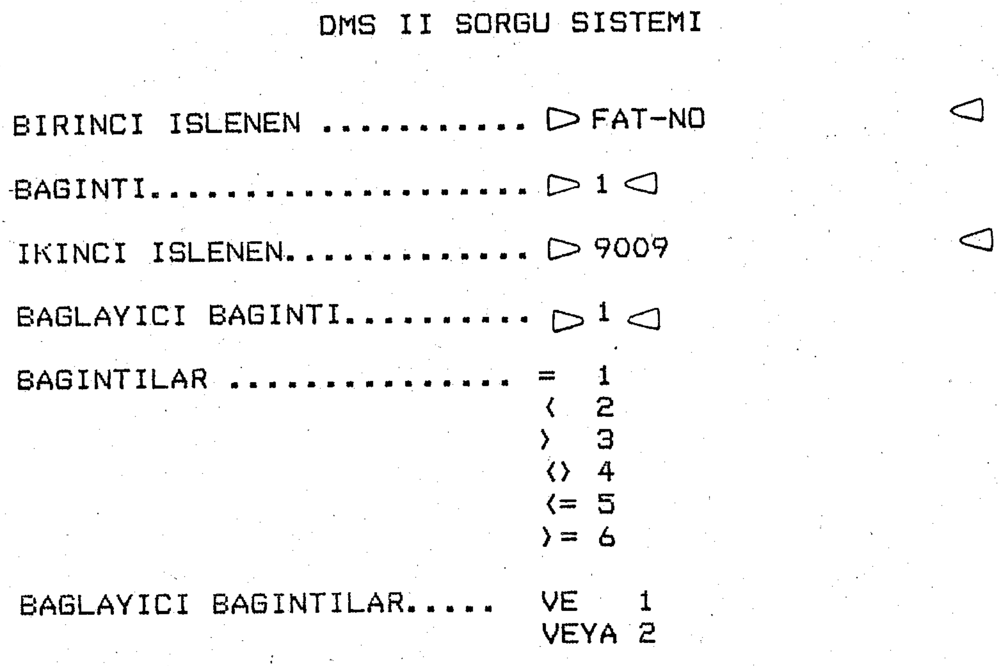
```
YAPI.........................  ▷FATURA          ◁
```

Option to update the database (4) is elected. The user elects to update a "FATURA" entry.



```
                    DMS II SORGU SISTEMI



YAPI.........................  ▷FATURA              ◁
```

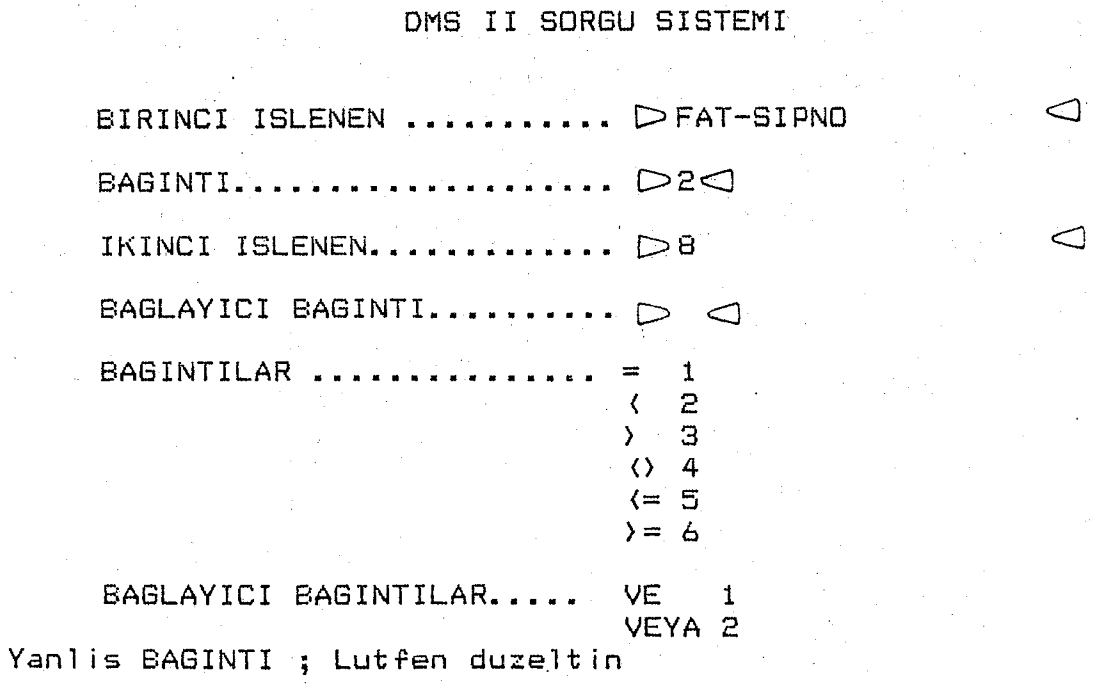
The value 9009 is assigned to the item FAT-NO



```
                    DMS II SORGU SISTEMI

BIRINCI ISLENEN ........... ▷FAT-NO                    ◁

BAGINTI.................... ▷ 1 ◁

IKINCI ISLENEN............. ▷ 9009                     ◁

BAGLAYICI BAGINTI.......... ▷ 1 ◁

BAGINTILAR ................ =   1
                            <   2
                            >   3
                            <>  4
                            <=  5
                            >=  6

BAGLAYICI BAGINTILAR.....   VE    1
                            VEYA  2
```

The value 8 should be assigned to the item FAT-SIPNO.

The user is not permitted to enter relations other than "=".



```
                    DMS II SORGU SISTEMI

BIRINCI ISLENEN ........... ▷FAT-SIPNO                 ◁

BAGINTI.................... ▷2◁

IKINCI ISLENEN............. ▷8                         ◁

BAGLAYICI BAGINTI.......... ▷  ◁

BAGINTILAR ................ =   1
                            <   2
                            >   3
                            <>  4
                            <=  5
                            >=  6

BAGLAYICI BAGINTILAR.....   VE    1
                            VEYA  2
Yanlis BAGINTI ; Lutfen duzeltin
```

The connecting relation should not be "VEYA"

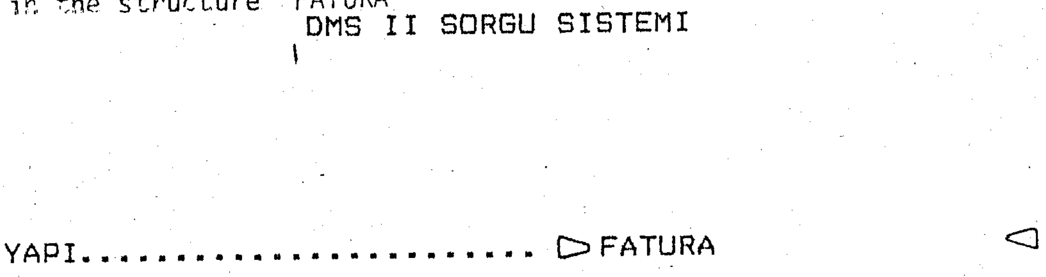
```
                          DMS II SORGU SISTEMI

    BIRINCI ISLENEN ........... ▷FAT-SIPNO                    ◁

    BAGINTI.................... ▷1 ◁

    IKINCI ISLENEN............. ▷8                            ◁

    BAGLAYICI BAGINTI.......... ▷2 ◁

    BAGINTILAR ............... =   1
                               <   2
                               >   3
                               <>  4
                               <=  5
                               >=  6

    BAGLAYICI BAGINTILAR.....  VE   1
                               VEYA 2
Yanlis BAGLAYICI BAGINTI ;Lutfen duzeltin
```

After correcting the errors the query was executed.

```
    BIRINCI ISLENEN ........... ▷FAT-SIPNO                    ◁

    BAGINTI.................... ▷1 ◁

    IKINCI ISLENEN............. ▷8                            ◁

    BAGLAYICI BAGINTI.......... ▷  ◁

    BAGINTILAR ............... =   1
                               <   2
                               >   3
                               <>  4
                               <=  5
                               >=  6

    BAGLAYICI BAGINTILAR.....  VE   1
                               VEYA 2
Yanlis BAGLAYICI BAGINTI ;Lutfen duzeltin


#
```

"#" stands for the successful completion.

OPTION next (b) was elected at the main menu. The user elects the next record in the structure "FATURA"

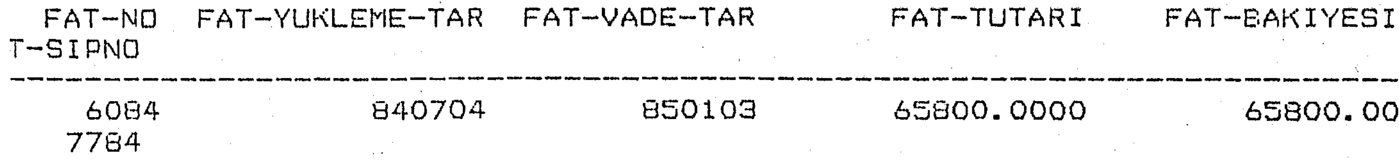


```
                  DMS II SORGU SISTEMI




YAPI........................ ▷FATURA                    ◁
```

Option DISPLAY (2) was elected at the main menu.

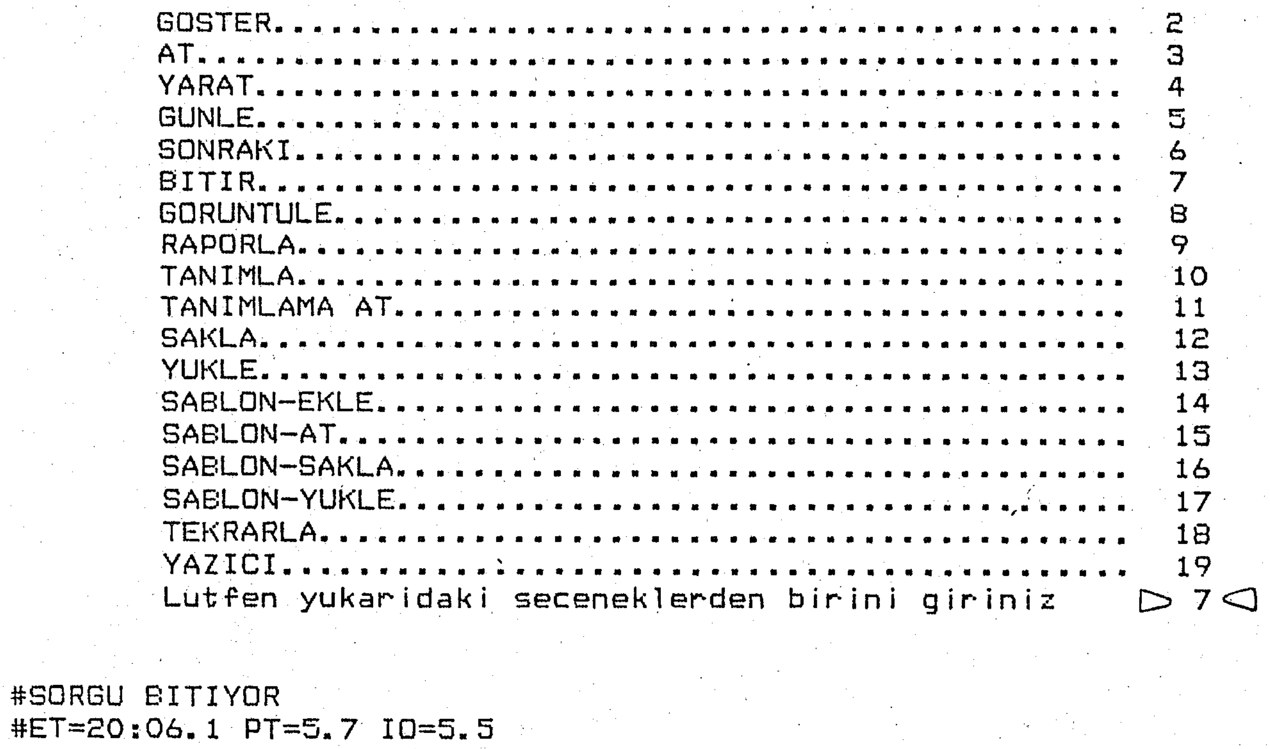
```
YAPI........................
```

The record is displayed.



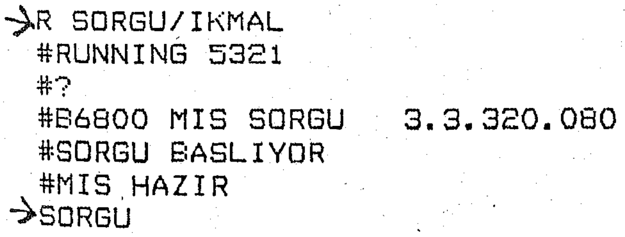
```
  FAT-NO  FAT-YUKLEME-TAR  FAT-VADE-TAR        FAT-TUTARI     FAT-BAKIYESI
T-SIPNO
-----------------------------------------------------------------------------
    6084            840704         850103       65800.0000           65800.00
   7784
```

The SORGU session was terminated after electing (7) at the main menu.



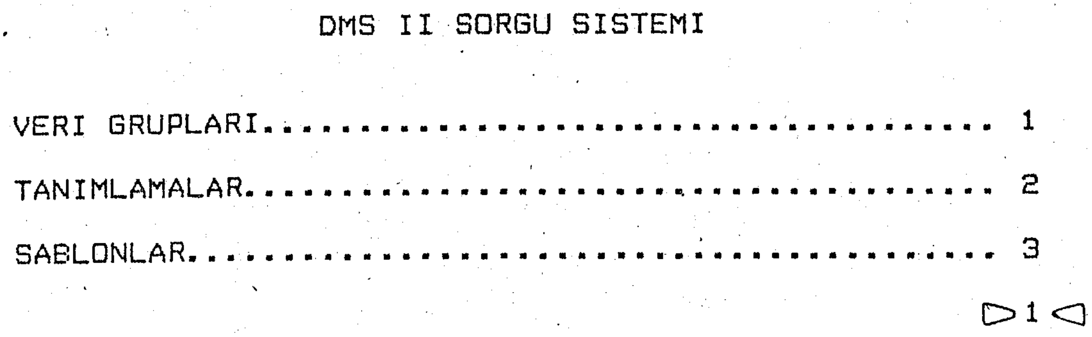
```
        GOSTER........................................   2
        AT............................................   3
        YARAT.........................................   4
        GUNLE.........................................   5
        SONRAKI.......................................   6
        BITIR.........................................   7
        GORUNTULE.....................................   8
        RAPORLA.......................................   9
        TANIMLA.......................................  10
        TANIMLAMA AT..................................  11
        SAKLA.........................................  12
        YUKLE.........................................  13
        SABLON-EKLE...................................  14
        SABLON-AT.....................................  15
        SABLON-SAKLA..................................  16
        SABLON-YUKLE..................................  17
        TEKRARLA......................................  18
        YAZICI........................................  19
        Lutfen yukaridaki seceneklerden birini giriniz   ▷ 7◁


#SORGU BITIYOR
#ET=20:06.1 PT=5.7 IO=5.5
```

Another SORGU session beging.



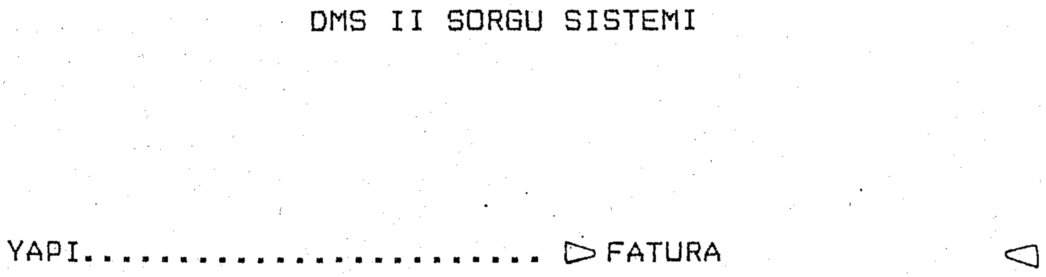
```
→R SORGU/IKMAL
 #RUNNING 5321
 #?
 #B6800 MIS SORGU   3.3.320.080
 #SORGU BASLIYOR
 #MIS HAZIR
→SORGU
```

Option Show (8) was elected at the main menu. The user responds to see the data sets of the data base by electing "1" at this menu.



```
                    DMS II SORGU SISTEMI

VERI GRUPLARI...................................... 1

TANIMLAMALAR....................................... 2

SABLONLAR.......................................... 3

                                                  ▷1◁
```

The user enters the name of the data set.



```
                    DMS II SORGU SISTEMI



YAPI......................... ▷FATURA               ◁
```

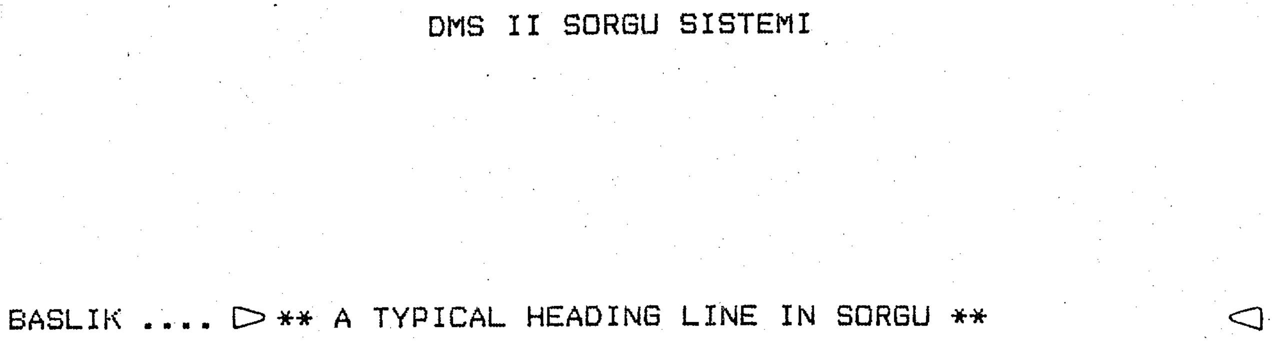
The item's in the data set is displayed.

```
FATURA VERI GRUBU
OGELER:
  FAT-NO NUMBER (10)
  FAT-YUKLEME-TAR NUMBER (6)
  FAT-VADE-TAR NUMBER (6)
  FAT-TUTARI NUMBER (14,4)
  FAT-BAKIYESI NUMBER (S12,2)
  FAT-SIPNO NUMBER (7)
GRUPLAR:
  FATNO KEYS FAT-SIPNO, FAT-NO
  FATTARIH KEY FAT-VADE-TAR
#
```

Option Report (9) was elected at the main menu. The structure that will be used is "FATTARIH".

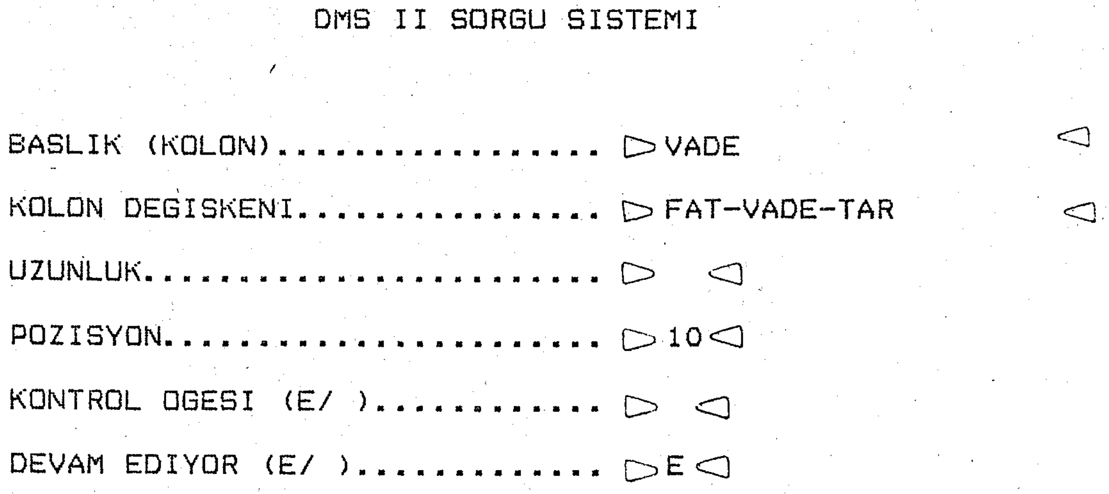
```
                         DMS II SORGU SISTEMI



     YAPI......................... ▷FATTARIH                 ◁
```

The heading of the report is entered.



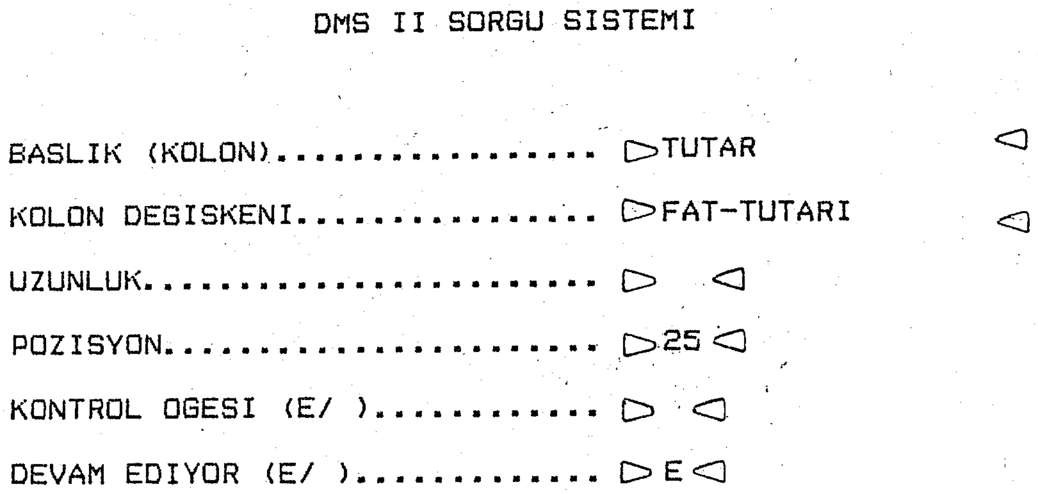
```
                     DMS II SORGU SISTEMI




BASLIK .... ▷** A TYPICAL HEADING LINE IN SORGU **            ◁
```

The first element of the report is entered.



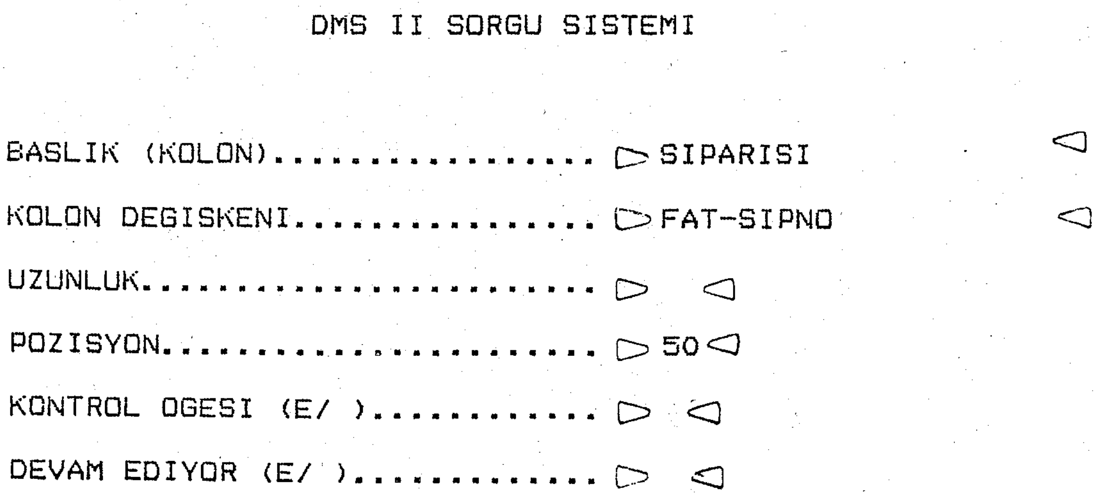
```
                  DMS II SORGU SISTEMI


BASLIK (KOLON).................. ▷VADE               ◁

KOLON DEGISKENI................. ▷FAT-VADE-TAR       ◁

UZUNLUK......................... ▷  ◁

POZISYON........................ ▷10◁

KONTROL OGESI (E/ )............. ▷  ◁

DEVAM EDIYOR (E/ )............. ▷E ◁
```

The second element of the report is entered.



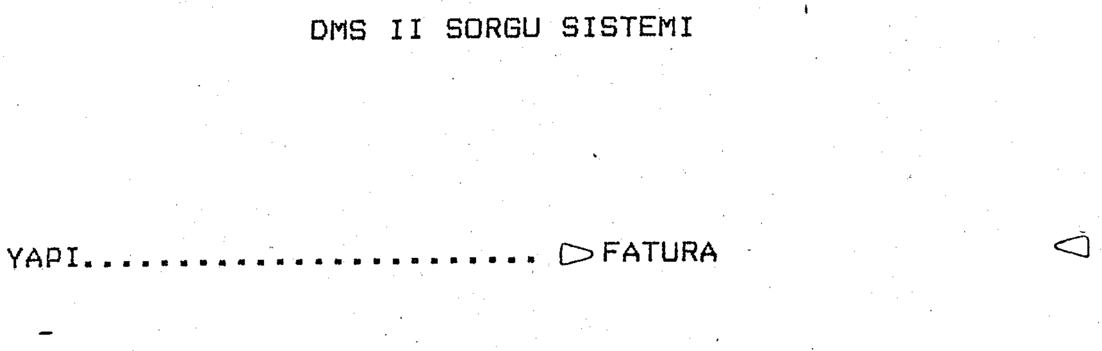
```
                    DMS II SORGU SISTEMI


BASLIK (KOLON)................. ▷TUTAR              ◁

KOLON DEGISKENI................ ▷FAT-TUTARI         ◁

UZUNLUK........................ ▷   ◁

POZISYON....................... ▷25 ◁

KONTROL OGESI (E/ )............ ▷  ◁

DEVAM EDIYOR (E/ )............. ▷E◁
```

The third and 2 the list element of the report is entered.



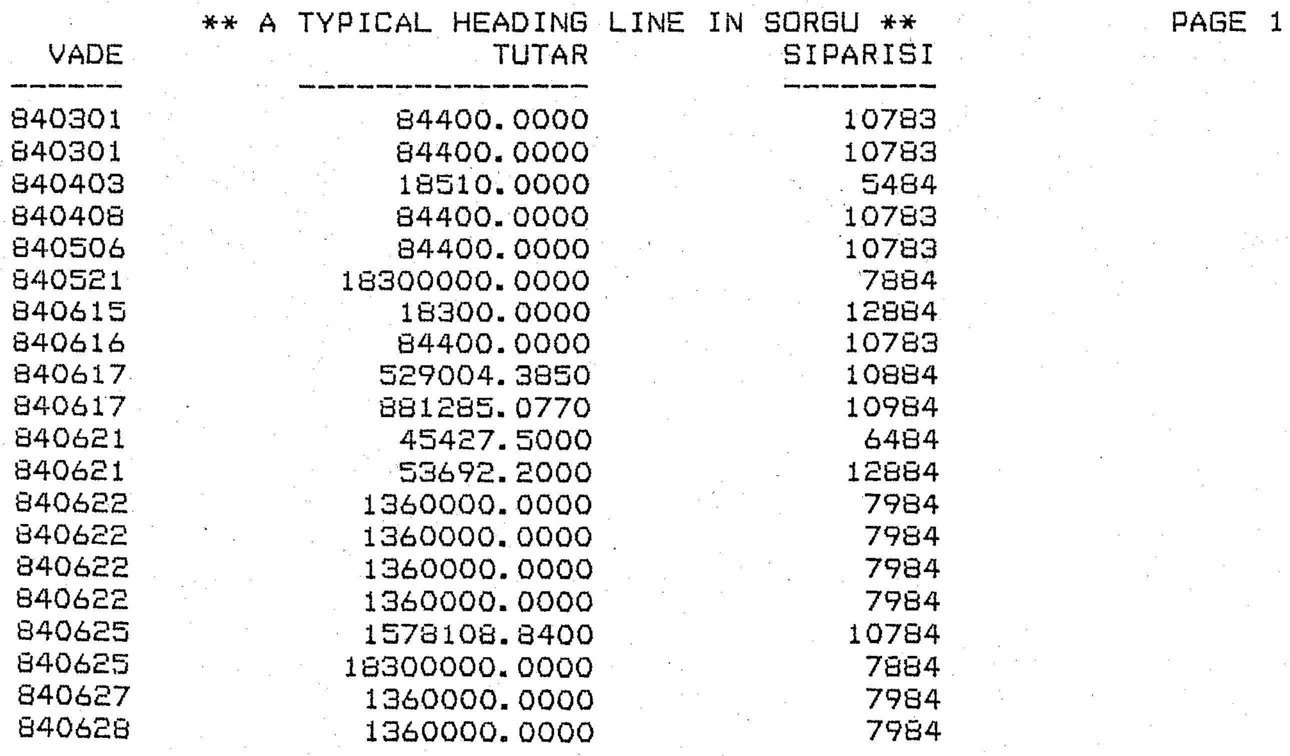
```
                    DMS II SORGU SISTEMI


BASLIK (KOLON)................. ▷ SIPARISI             ◁

KOLON DEGISKENI................ ▷FAT-SIPNO            ◁

UZUNLUK........................ ▷    ◁

POZISYON....................... ▷ 50◁

KONTROL OGESI (E/ )............ ▷  ◁

DEVAM EDIYOR (E/ )............. ▷  ◁
```

The report will be produced using the "FATURA" data set.



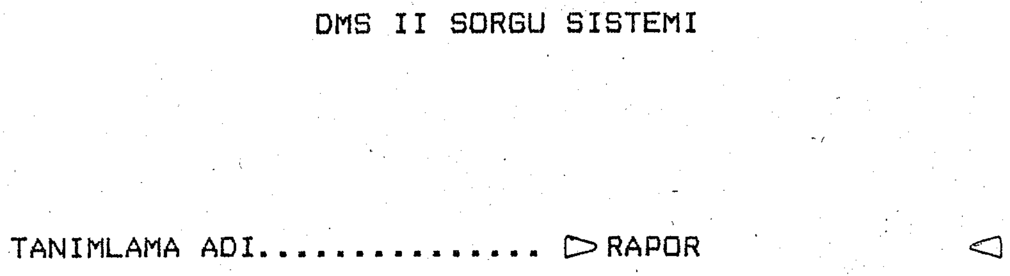
```
                 DMS II SORGU SISTEMI



YAPI........................ ▷FATURA                   ◁

  _
```

The report is displayed.



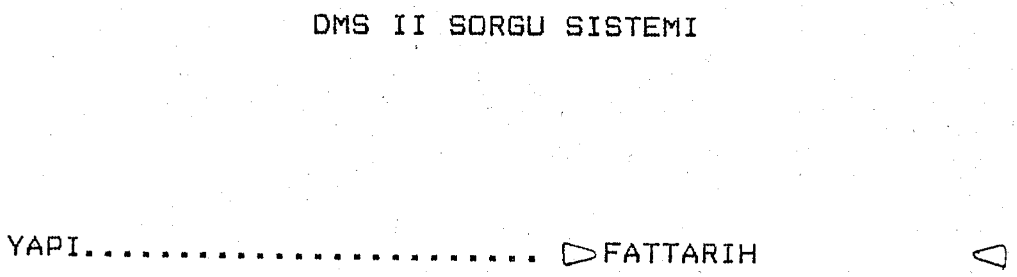
```
           ** A TYPICAL HEADING LINE IN SORGU **             PAGE 1
  VADE                     TUTAR            SIPARISI
 ------            ---------------          --------
 840301                84400.0000             10783
 840301                84400.0000             10783
 840403                18510.0000              5484
 840408                84400.0000             10783
 840506                84400.0000             10783
 840521             18300000.0000              7884
 840615                18300.0000             12884
 840616                84400.0000             10783
 840617               529004.3850             10884
 840617               881285.0770             10984
 840621                45427.5000              6484
 840621                53692.2000             12884
 840622              1360000.0000              7984
 840622              1360000.0000              7984
 840622              1360000.0000              7984
 840622              1360000.0000              7984
 840625              1578108.8400             10784
 840625             18300000.0000              7884
 840627              1360000.0000              7984
 840628              1360000.0000              7984
```

Option Define (10) was elected at the main menu. The define name is "RAPOR".



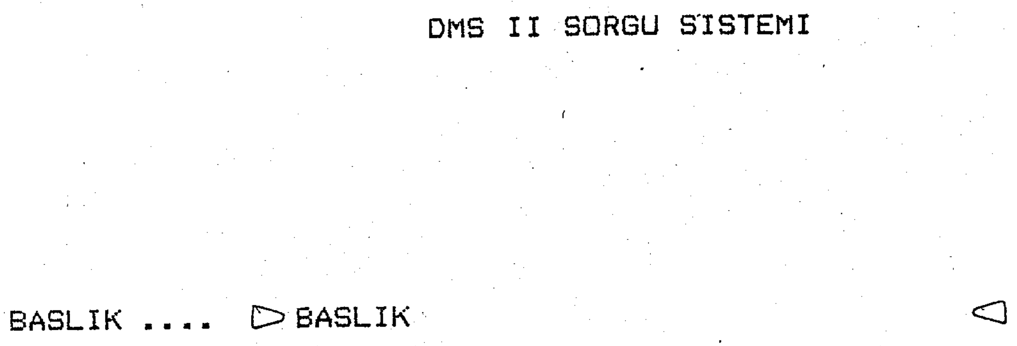
```
                    DMS II SORGU SISTEMI




TANIMLAMA ADI............... ▷RAPOR                   ◁
```

The structure that will be used is "FATTARIH".



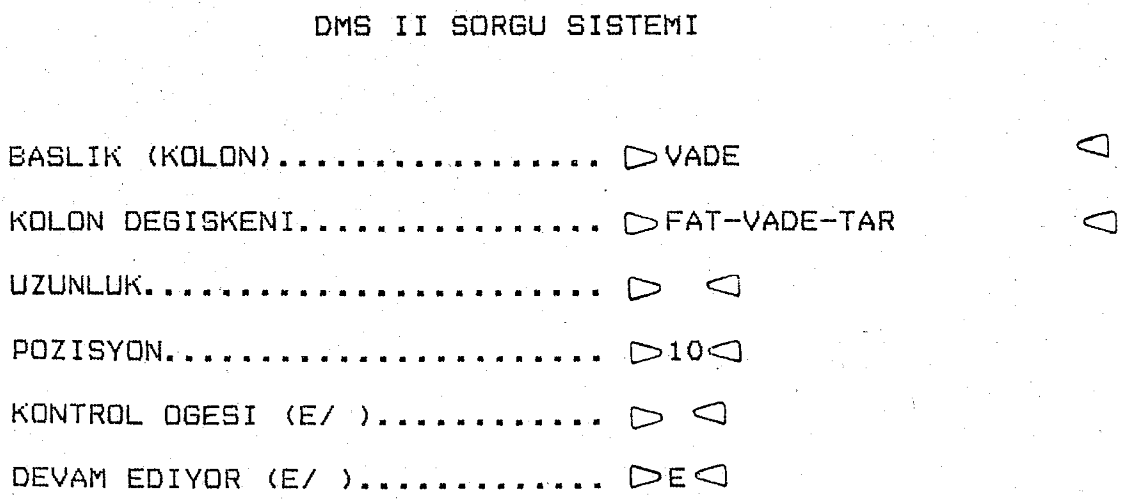
```
                    DMS II SORGU SISTEMI




YAPI........................ ▷FATTARIH                ◁
```

The heading of the renort is entered.



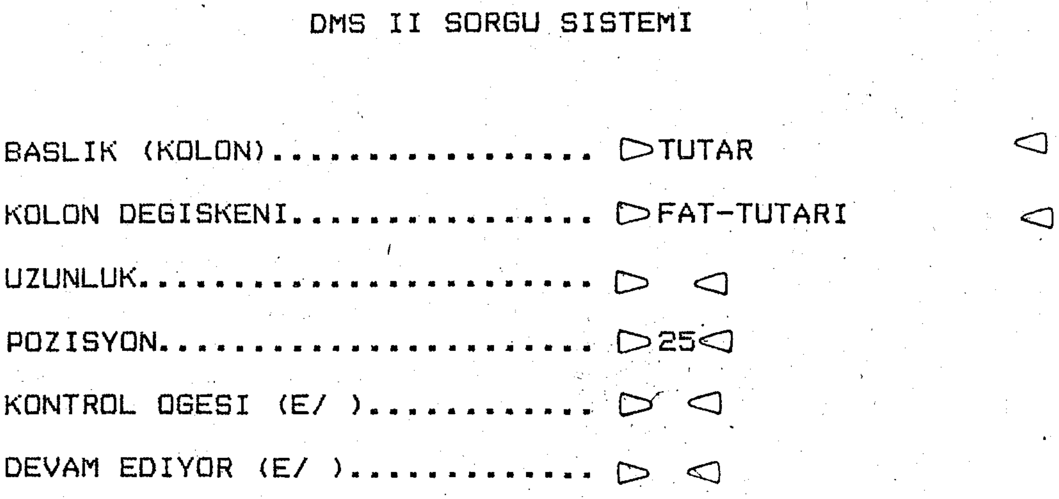
```
                         DMS II SORGU SISTEMI




BASLIK ....   ▷BASLIK                                  ◁
```

The first element of the report is "FAT-VADE-TAR".



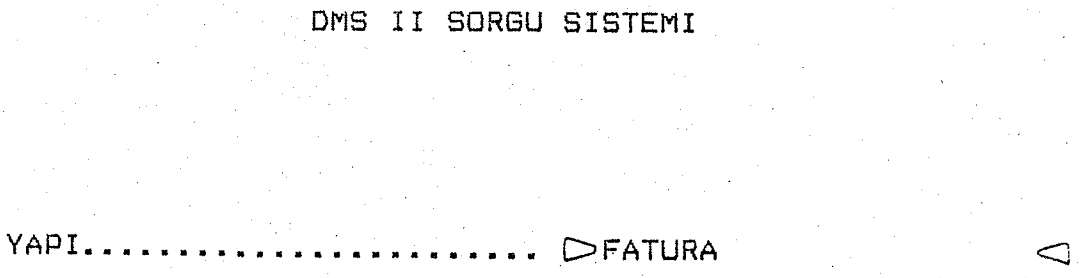
```
                         DMS II SORGU SISTEMI


BASLIK (KOLON)..................  ▷VADE                          ◁

KOLON DEGISKENI.................  ▷FAT-VADE-TAR                  ◁

UZUNLUK.........................  ▷   ◁

POZISYON........................  ▷10◁

KONTROL OGESI (E/ )............   ▷  ◁

DEVAM EDIYOR (E/ ).............   ▷E◁
```

The second element of the report is "FAT-TUTARI".



```
                    DMS II SORGU SISTEMI


BASLIK (KOLON).................. ▷TUTAR                 ◁

KOLON DEGISKENI................. ▷FAT-TUTARI            ◁

UZUNLUK......................... ▷   ◁

POZISYON........................ ▷25◁

KONTROL OGESI (E/ )............. ▷   ◁

DEVAM EDIYOR (E/ )............... ▷   ◁
```

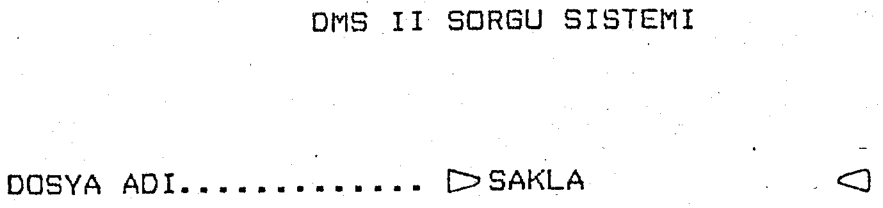
The report will be generated using the data set "FATURA".



```
                    DMS II SORGU SISTEMI



YAPI.......................... ▷FATURA                 ◁
```

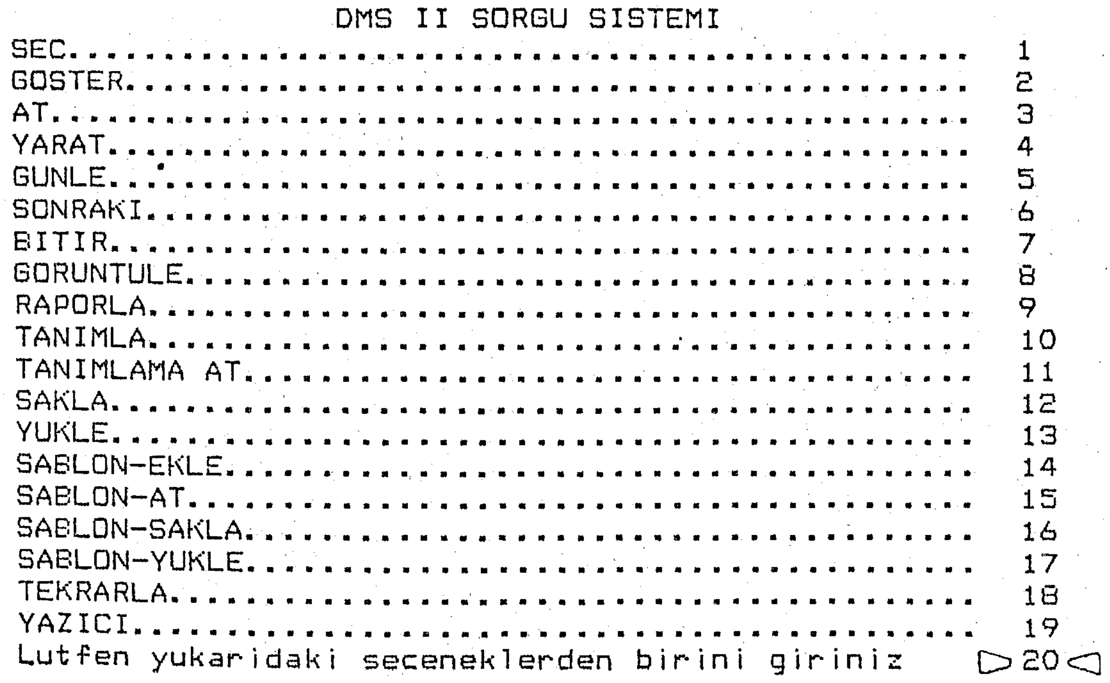
" " sign indicates successful competion.

Option SAKLA (12) was elected at the main menu. The dump file name is "SAKLA"



```
                    DMS II SORGU SISTEMI



DOSYA ADI............. ▷SAKLA                ◁
```

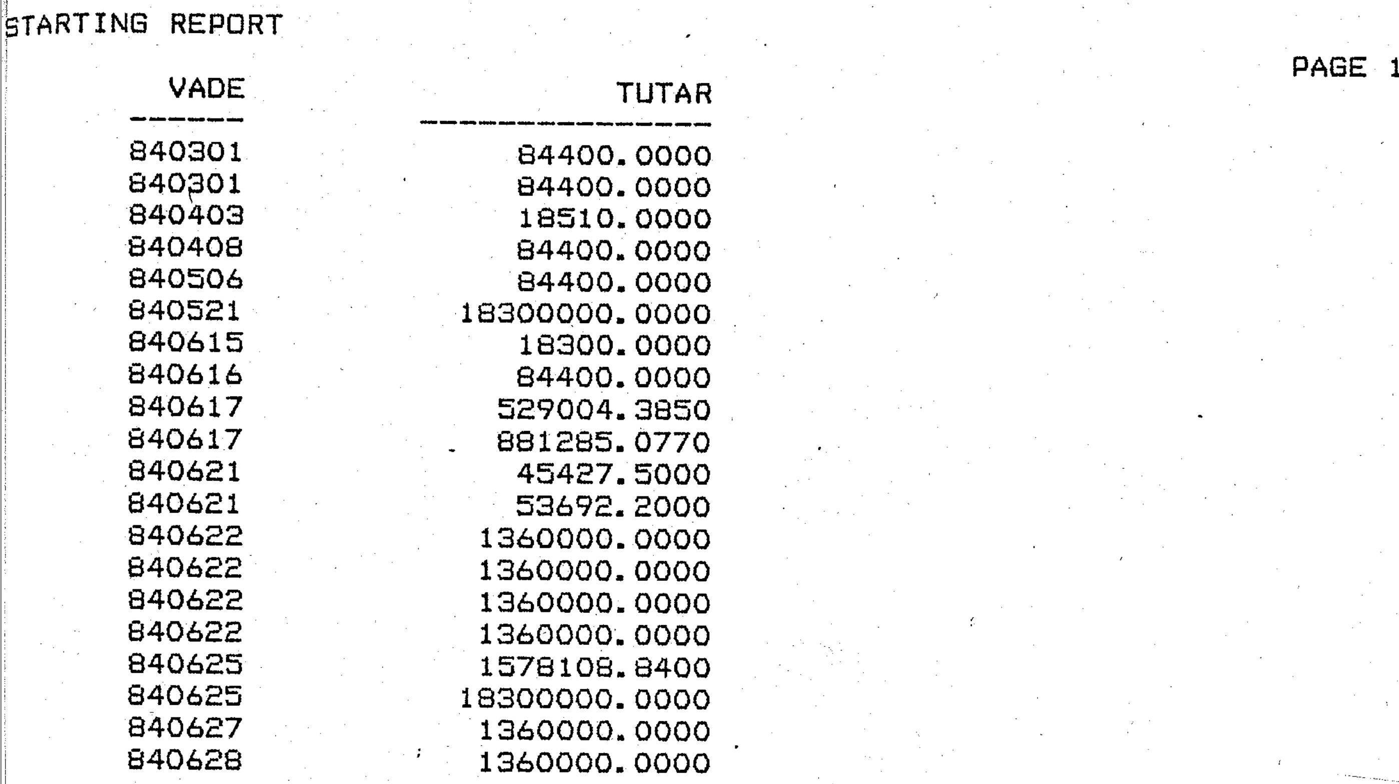
The user exists the SORGU phase by electing "20" at the main menu.



```
                     DMS II SORGU SISTEMI
SEC................................................    1
GOSTER.............................................    2
AT.................................................    3
YARAT..............................................    4
GUNLE..............................................    5
SONRAKI............................................    6
BITIR..............................................    7
GORUNTULE..........................................    8
RAPORLA............................................    9
TANIMLA............................................   10
TANIMLAMA AT.......................................   11
SAKLA..............................................   12
YUKLE..............................................   13
SABLON-EKLE........................................   14
SABLON-AT..........................................   15
SABLON-SAKLA.......................................   16
SABLON-YUKLE.......................................   17
TEKRARLA...........................................   18
YAZICI.............................................   19
Lutfen yukaridaki seceneklerden birini giriniz      ▷20◁
```

› RAPOR

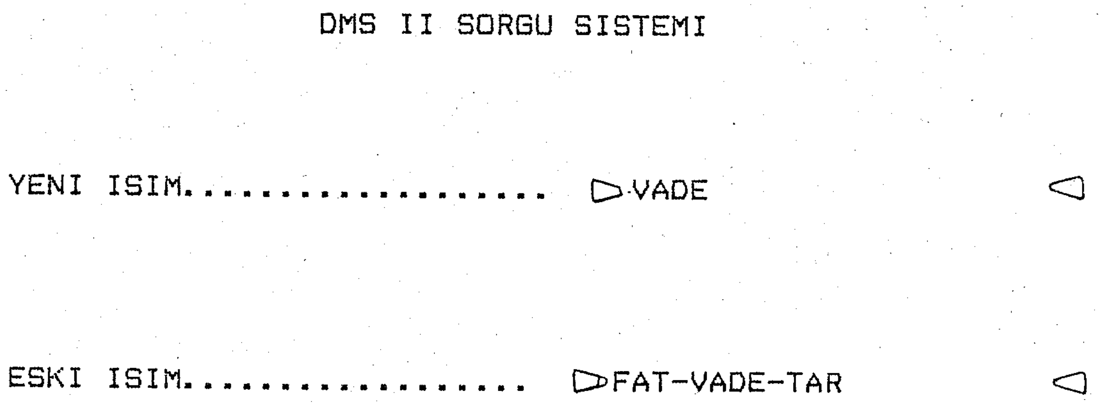
The define item "RAPOR" was involved.



```
STARTING REPORT                                                           PAGE 1

     VADE                    TUTAR
    ------              --------------
    840301                84400.0000
    840301                84400.0000
    840403                18510.0000
    840408                84400.0000
    840506                84400.0000
    840521             18300000.0000
    840615                18300.0000
    840616                84400.0000
    840617               529004.3850
    840617               881285.0770
    840621                45427.5000
    840621                53692.2000
    840622              1360000.0000
    840622              1360000.0000
    840622              1360000.0000
    840622              1360000.0000
    840625              1578108.8400
    840625             18300000.0000
    840627              1360000.0000
    840628              1360000.0000
```

The report was generated after entering the define item "RAPOR".

Option Insert mapping relation (14) was elected at the main menu. The name "VADE" will be used instead of "FAT-VADE-TAR".



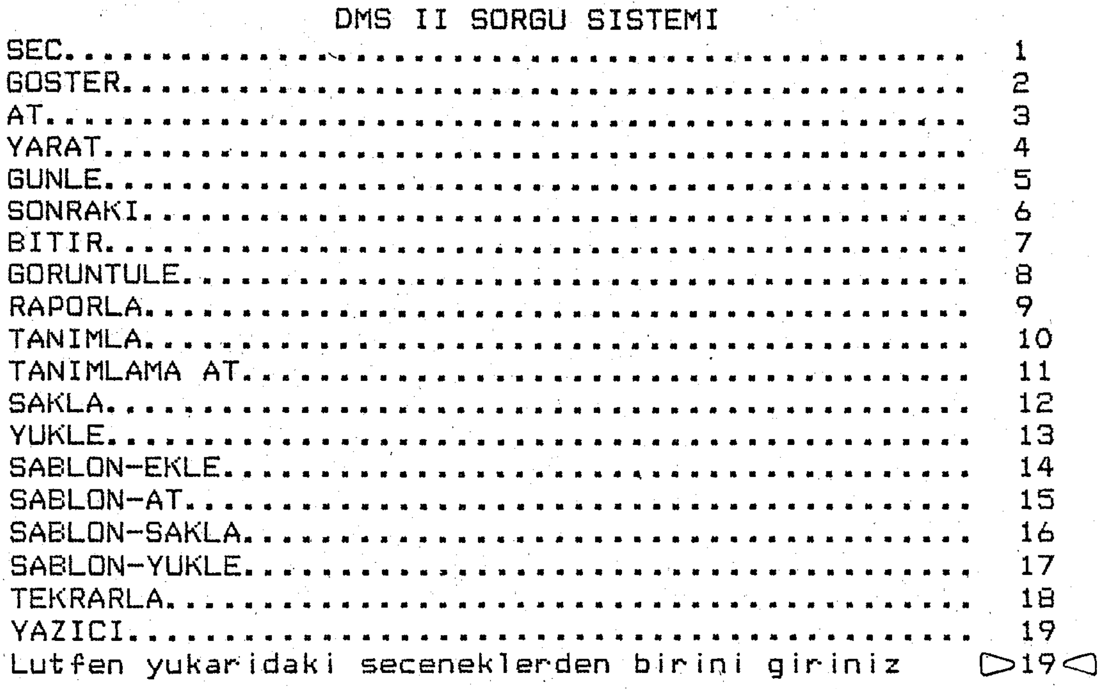
```
                    DMS II SORGU SISTEMI


YENI ISIM..................   ▷VADE                    ◁


ESKI ISIM..................  ▷FAT-VADE-TAR             ◁
```

Option Show (8) was elected at the main menu. The user wishes to see the mapping relations.

```
                    DMS II SORGU SISTEMI

VERI GRUPLARI.................................... 1

TANIMLAMALAR..................................... 2

SABLONLAR........................................ 3
                                                ▷3◁
```

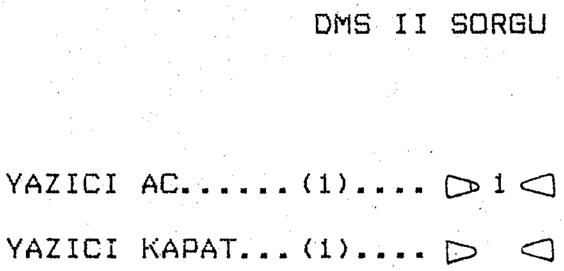
The mapping relations are displayed.

```
VADE                                  --)   FAT-VADE-TAR
F                                     --)   FATURA
```



```
                    DMS II SORGU SISTEMI
SEC............................................   1
GOSTER.........................................   2
AT.............................................   3
YARAT..........................................   4
GUNLE..........................................   5
SONRAKI........................................   6
BITIR..........................................   7
GORUNTULE......................................   8
RAPORLA........................................   9
TANIMLA........................................   10
TANIMLAMA AT...................................   11
SAKLA..........................................   12
YUKLE..........................................   13
SABLON-EKLE....................................   14
SABLON-AT......................................   15
SABLON-SAKLA...................................   16
SABLON-YUKLE...................................   17
TEKRARLA.......................................   18
YAZICI.........................................   19
Lutfen yukaridaki seceneklerden birini giriniz   ▷19◁
```

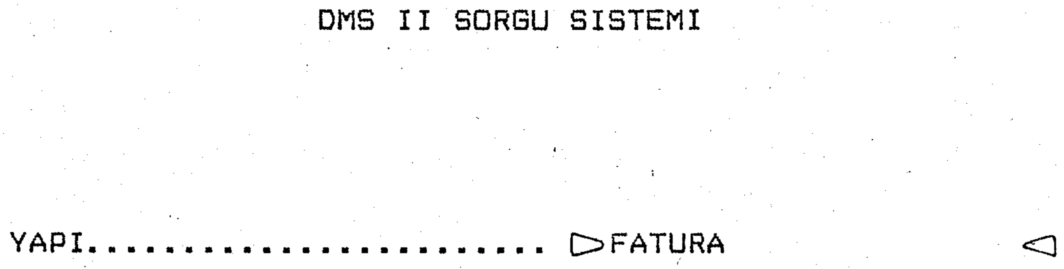
"19" was entered at the main menu for settinn the printer option.



```
                    DMS II SORGU SISTEMI


YAZICI AC......(1).... ▷1◁

YAZICI KAPAT...(1).... ▷  ◁
```

The print ontion was set.

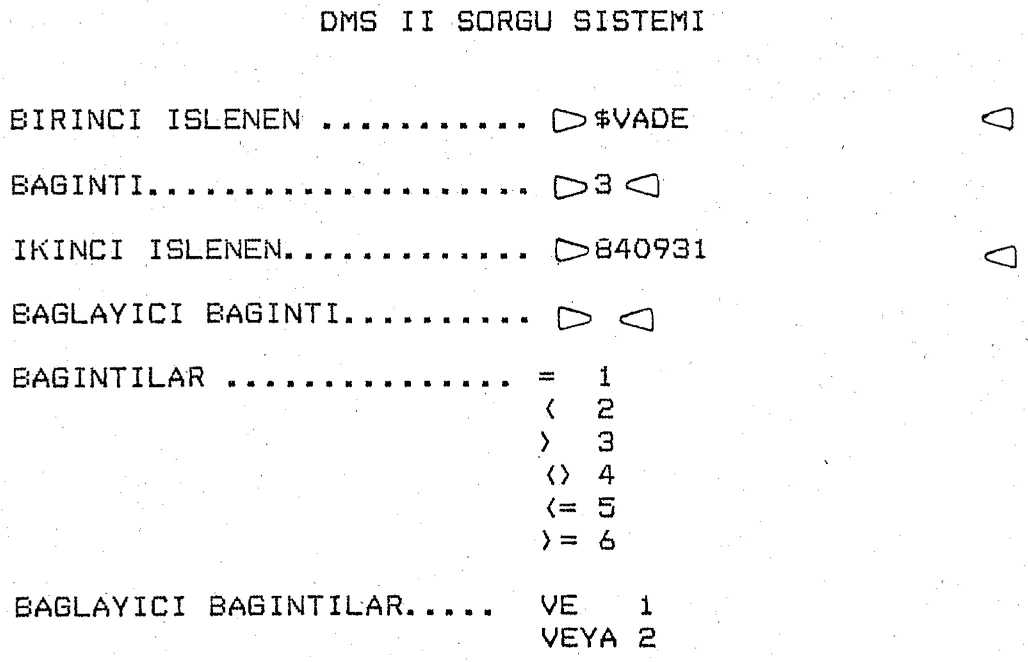
Option Select (1) was elected at the main menu.



```
                    DMS II SORGU SISTEMI




YAPI......................... ▷FATURA                         ◁
```

Structure FATURA is elected.

Search criteria is defined as FAT-VADE-TAR (VADE) 840931



```
                    DMS II SORGU SISTEMI

BIRINCI ISLENEN ........... ▷$VADE                        ◁

BAGINTI.................... ▷3 ◁

IKINCI ISLENEN............. ▷840931                       ◁

BAGLAYICI BAGINTI.......... ▷ ◁

BAGINTILAR ................ =   1
                            <   2
                            >   3
                            <>  4
                            <=  5
                            >=  6

BAGLAYICI BAGINTILAR.....   VE    1
                            VEYA  2
```

#

"#" sign implies successful completion.

Option Display (2) was elected at the main menu.

```
        YAPI.........................
```

```
# DISPLAY # 1 ON PRINTER
#
```

Instead of displaying the record the hardcopy output is produced.

Option repeat (18) was elected at the main menu.

```
        YAPI.........................
```

```
# DISPLAY # 2 ON PRINTER
#
```

Since no structure is specified the last commond was reexecuted.

IV. DESCRIPTION OF SORGU

SORGU has been written in DMALGOL which is a programming language that has data base management function extensions of standard ALGOL. SORGU software comprises two major components i) The extensions in BUILDINQ, ii) The extensions in INQUIRY program.

The extensions in BUILDINQ.

BUILDINQ is a relatively small program (compared to INQUIRY) that passes the related parameters and forms the DMINQDIRECTORY, and also if specified zips the INQUIRY compile. Only the messages of the BUILDINQ has been translated to Turkish and the related patch can be seen in appendix.

Te extensions in INQUIRY.

INQUIRY is the main interactive program that handles the queries. Again some messages are translated to Turkish in this program. But the main objective was the menu-driven query forming feature of the SORGU. Only a small portion of the original program have been changed to branch to the SORGU features.

Before going into further details of the SORGU I should point out that while Burroughs deals with the INQUIRY software SORGU can be used anytime. In other words the patches have been prepared so that when the MCP operating system changes levels SORGU will not need any modifications. This is an important feature because every year Burroughs releases a new version of the MCP operating system.

4.1. Characteristics of SORGU

Although the INQUIRY serves as a good tool for accessing the data in a DMS-II database, there are several disadvantages. First of all the user should know English since all the commands and replies are in english. INQUIRY can be used easily by the persons who have a notion of the computer and its use, but people who are unfamiliar with the computer can not form the queries with INQUIRY easily.

SORGU the Turkish subset of the INQUIRY, was designed to overcome these difficulties. It was designed so that the user having no idea about the computer can easily use the data base and form his queries. A menu-driven nature has been added to SORGU in order to increase the ease of use. SORGU is a subset of INQUIRY because only the most important commands of INQUIRY are included.SORGU makes a preliminary pass over the query forming menu in order to form true queries for the INQUIRY but most semantic error checking is done by the INQUIRY. Since the semantic checking is done by the INQUIRY the relative error messages have been translated to Turkish.

SORGU is embedded into the INQUIRY so that one can use INQUIRY facilities as well as the SORGU facilities. The user can get into the INQUIRY and form his queries and whenever he wants to use the SORGU he only types "SORGU" and the turkish part of the software is ready for use. This facility is useful for professional users of SORGU because they can immediately use the parts of INQUIRY excluded in SORGU. For example the user can GENERATE a temporarry set first and then using SORGU the queries can be formed easily.

4.2. Procedures of SORGU

4.2.1. Screen Handling

The screen handling of SORGU has been implemented using the "TD830SUPPORT" feature of the MCP operating system. The TD830 is a general term of the terminals in the classes TD830, MT983, MT985, and the ET1100 series of terminals. The ET1100 series of terminals are the most ercently released type of terminals (autumn 1983) and SORGU was designed on this series of terminals.

The procedure CURSOR with its three parameters row number, column number, and pointer to buffer; locetes the pointer to the specified row and column.

The procedure WRYTE displays the screen pointed by its second parameter through the file specified in its first parameter.

The procedure NFORMS contains the menu and screens of SORGU. It has only one parameter which is used for determining which screen will be displayed. The user DEFINEd verb SAY places the trailing text in positions pointed by pointer PBUF. In general, every screen format begins with the hexadecimal character "DC" for clearing the screen then the function CURSOR places the pointer and SAY puts the given text in it. At the end there are the hexadecimal characters "27E603"; 27 stands for escape character, E6 stands for putting the screen into forms mode, and 03 stands for end of the message. The forms mode is a useful concept in Burroughs machines you can get only the information within the delimiters ("▷", "△", and "◁") as input from the whole screen.

4.2.2. Screens of SORGU

The screens of SORGU can be listed as follows:

0: The main menu of SORGU.

1: The screen for the structure name.

2. The screen for forming the conditions.

3: The screen for the heading of a report.

4: The screen for the items in a report list.

5: The screen for the mapping relations.

6: The screen for the file name,

7: The screen for the define-name.

8: The screen that identifies the options of the command GÖRÜNTÜLE.

9: The screen for setting the printer options.

These are the general screens of SORGU, there are also special message output using the procedures CURSOR and WRYTE. On these messages (various places in the program) there is only one difference the hexadecimal character "OC" is not used therefore the screen is not cleared but the hexadecimal character "27E3" is used instead for puttin the screen into its original position. In Burroughs large systems the screen is scrolled up while in input or output. In order to overcome this side effect the screen is scrolled down one line for gaining the original position.

4.2.3. Main Body

The main body can be examined in two parts; the procedures that are appended to INQUIRY, and the lines that are appended to original INQUIRY source.

-Procedures appended to INQUIRY.

The procedure NMAIN is the main controlling procedure of SORGU. First it displays the main menu and according to the input passes control to the appropriate procedure. The inputs can be:

1 SEÇ for selecting,

2 GÖSTER for displaying,

3 AT for deleting,

4 YARAT for creating,

5 GÜNLE for updating,

6 SONRAKİ for next,

7 BİTİR to end the session,

8 GÖRÜNTÜLE to show the database items, defined items, or mappings,

9 RAPORLA for reporting,

10 TANIMLA for defining permanent reports,

11 TANIMLA AT for deleting previously defined reports,

12 SAKLA for saving the defined reports,

13 YÜKLE for loading the previously defined reports,

14 ŞABLON EKLE for inserting new mapping relations.

15 ŞABLON AT for deleting the previously defined mapping relations,

16 ŞABLON SAKLA for saving the mappings,

17 ŞABLON YÜKLE for loading the previously defined mappings,

18 TEKRARLA for repeating the last command,

19 YAZICI for setting the printer attributes,

20 for barnching to the original INQUIRY software.

There are twelve operational procedures that control passes from NMAIN. These are

- NSELECTDISPLAYP,
- NDELETCREATEUPDATEP,
- NQUITNEXTP,
- NSHOWP,

- NREPORTP,
- NDEFINEP,
- NCLEARP,
- NSAVERESTOREP,
- NTANIMLAP,
- NTANIMLAATP,
- NMAPSAVERESTOREP,
- NPRINTERP.

The commands are grouped so that similar functions are done in only one procedure.

These operational procedures form the queries and write the queries into a temporary disk file called "DISKIM". INQUIRY read these queries from disk instead of terminal. Every time SORGU performs a write operation it increments the variable KAYITSAYISI by 1 and when INQUIRY processes an input it decrements this value so that it can determine when to read from disk or pass control to SORGU. When control passes from INQUIRY to SORGU the temporary file is purged and SORGU creates a new version of the file.

The procedure NSELECTDISPLAYP forms the queries for the commands "SEÇ" and "GÖSTER". First it displays the screen for inputting the structure needed. If the user does not specify any structure this means that the user has changed his mind and wants to return to the main menu. This is done by setting the variable GERIYEDON to 1. After inputting the structure name the user is asked for the relations. There are four fields for inputing values; these are first operand, operator, second operator, and attaching operator. Usually the first operand is an item name in the specified structure and the operator is a relational operator from one of the following "=", "<", ">", "<>", "<=", ">=". The second operand can be an item in the data base or a value for the first operator. The attaching operator can be "ve" or "veya" and is used for forming queries that are based on more than one condition.

Basically a condition is formed by the first and second operands and the operator. If the attaching operator does not exist it means that the query is finished and control is passed from this procedure to NMAIN. There is a limit in attaching these conditions by "ve" or "veya" it is 47. There exists numeric codes for operators and attaching operators and the information about them is listed on the screen. The procedure also checks whether the operator is one of the ("=", "<", ">", "<>", "<=", ">=") and as mentioned before the attaching operators can be "ve" or "veya" if this is not the case error messages are displayed. This procedure also handles an exceptional case whereby the user may select a structure with a conditional expression; when a record fits the criteria " " is displayed then the user should enter the command "GÖSTER" and in stead of the structure name he should enter a null string. SORGU will display the record using this command. While using the commands "SEÇ" and "GÖSTER" if there is no record that satisfies the conditional expression the message "YOK" will be displayed.

The procedure NDELETECREATEUPDATEP handles the commands "AT", "YARAT", and "GÜNLE". The nature of this procedure is similar to NSELECTDISPLAYP. The commands "AT" and "GÜNLE" require only a structure name for forming the query; therefore if a structure name is inputted then the query is ready. Again if no structure name is specified then control is passed to the main menu. For the command "YARAT" the user should assign values to items in the dataset in order to create a record of the data set. The conditioning screen again appears and the user should input the operator "=", the item, value for the item, and if the list of items are not exhausted "VE" for continuation. Since this is a creation command the user can not specify any other relational operator but "="; otherwise it would be meaningless. If any other relational operator is encountered then an error message is displayed. Also the attaching operator "VEYA" can not be used because we are not forming conditions however, if the user enters this option then an error message is displayed.

In general, when an error message is displayed the user is asked to correct his fault and then the query formation process is continued.

The procedure NQUITNEXTP handles the commands "SONRAKİ" and "BİTİR". The command "BİTİR" does not need any additional information and it's function is to terminate SORGU. "SONRAKİ" is used for accessing to the next record that satisfies the most current criteria for that structure. If a structure is not specified for this command then the next record in the last accessed structure is displayed or selected. The user can also specify a structure and in that case the next record in that structure is accessed. This command should be used after "SEC" or "GÖSTER" commands and it does the same actions as it's previous command (i.e. selects or displays.). In all cases if there is no next record for that structure the message "DAHA YOK" is displayed.

The procedure NSHOWP is used for handling the command "GÖRÜNTÜLE". The function of this command is to show either data set names or the items in a specified data set including sets, subsets and their keys. If the user does specify a data set name then the items of the data set are displayed otherwise the data set names of the data base is listed.

The procedure NREPORTP handles the command "RAPORLA". First it displays the screen for inputting the structure to be used. If no structure is given a return to main menu is performed. Then the screen for inputting the heading of the report is displayed. After the heading of the report is received the screen for inputting the column heading, item to be listed, length of the item, position of the item on the line, and the flag showing whether the item is a controlling item or not is displayed. If invalid input for these items are encountered related error messages are displayed.

This process is repeated until the user specifies that the item is the last one. Lastly the structure name that the report will be generated from is inputted.

The procedure NDEFINEP handles the command "TANIMLA". It displays the screen for inputting the define-name. After recieving the define-name the procedure passes control to the procedure NREPORTP for recieving the related information for the defined report.

The procedure NCLEARP is used for deleting a defined report from the define list. It handles the command "TANIMLA AT", it displays a screen for inputting the define-name, the query formation process is completed after the user enters the define-name.

The procedure NSAVERESTOREP handles the commands "SAKLA" and "YÜKLE". The main function of the procedure depends on the command selected dumping or loading the defined reports. It displays the screen for inputting the name of the file on which dump load will occur. After recieving the file name the query is formed depending on the command.

The next quadruple of procedures forms a completely new facility which is only available in SORGU against INQUIRY. It is the mapping function. The user can map the item names into a list of user-defined names. For example the structure name FATURALAR can be mapped and only F can be used instead. The program uses a file named MAPS for saving the mappings and uses a matrix (NU) as the work area of the mappings. The first column contains the user defined name and the second the original name of the item in the database.

The procedure NTANIMLAP is used for defining the mapping functions and is invoked by the command "ŞABLON-EKLE". It displays a screen that the user can iput both the user defined name and the original item name. The procedure checks whether the limit of 100 mappings is exceeded or not. Also a check is made for whether the user defined name has been used in other mappings or not. If these checks are completed successfully the mapping function is included into the list.

The procedure NTANIMLAATP is used for deleting previously defined mapping relations and is invoked by the command "ŞABLON AT". It resembles the previous procedure, it displays a screen for inputting the relation. If the input is valid then the mapping relation is deleted from the list.

The procedure NMAPSAVERESTOREP handles the commands "ŞABLON YÜKLE" and ŞABLON SAKLA". The mapping relations can be saved in a file called MAPS. The first record of this file shows the number of relations in the file. The procedure either stores or loads the mapping declarations. If the command is storing the declerations, the pointer of the mappings is written into the first record and then the declarations are written. The loading process is done by first reading the pointer and then moving the declerations into the matrix NU.

The procedure NPRINTERP handles the command "YAZICI". It is used for changing the status of the printing option. SORGU can give the results in hardcopy by using this command. It displays a screen where the user can enter which option he needs. After checking the given input the related query is formed.

4.2.4. General Procedures

There are also two procedures where the twelve operational procedure passes control frequently. These are NYAZARTIK and ARA. The procedure NYAZARTIK increments the global record to be processed count and writes whole or segment of the query that has been formed.

The procedure ARA is used for determining whether the mapped item is in the mapping list (NU) or not. There may exist two classes of mapped items; data base item names or operands. The data base item names may contain at most 17 characters while the other class may contain 23 characters. The procedure has 3 parameters; PARAM is the token to be tested.

SONUÇ is the value to be returned, and TIP is the class of the token. The mapped items are identified by the character "$" in the first character of the string. The procedure drops the first character and tries to match the token with the list of declerations. If a successful match is found the value 1 is returned through the parameter SONUÇ and the original text is returned through PARAM.

4.3. Modifications to INQUIRY

There is also a patch prepared for merging into the original INQUIRY source for branching to and from SORGU. The branching is controlled majorly by the variable NCONT. If NCONT is 1 it means that SORGU is active at that instance. The patch is merget with the procedure REMOTEINPUT which handles the input via terminal. The reason for only changing the original software in only one procedure is for compatibility. One can remember that while inputting through SORGU every time a record is written to disk the counter KAYITSAYISI is incremented therefore if this variable is not equal to zero then it means that we are in the SORGU phase. (the terms SORGU phase and INQUIRY phase will be used in order to differentiate the instances where one can enter SORGU commands and INQUIRY commands.) If KAYITSAYISI is greater then zero then it should be read from the disk file where the queries in the SORGU phase were written. After reading a record we should decrement the value of KAYITSAYISI. If the KAYITSAYISI is equal to zero then we have processed all of the queries prepared at the SORGU phase therefore we should purge the disk file. Since opening of the file is handled automatically purging does not have any side effects. If KAYITSAYISI is equal to zero we should check whether we are in the SORGU or INQUIRY phase by the variable NCONT. If we are in the SORGU phase we pass control to the procedure NMAIN. When control returns from NMAIN we lock (close with lock) the disk file. Afterwards, we read a record from this file and let the INQUIRY process it. If we are in the INQUIRY phase the segment has no effect. The second part of the patch handles the key to enter to the SORGU phase. It helps the INQUIRY to detect the keyword "SORGU" and when it detects the keyword it sets the variable NCONT to one. The user can change the state of SORGU phase by entering "20" in the main menu of SORGU phase.

4.4. Generation of SORGU

SORGU generation using a terminal.

SORGU can be generated through CANDE, by entering

```
RUN OBJECT/BUILDSORGU
```

BUILDSORGU will respond with the message

```
HANGİ VERİ TABANI?
```

The user should enter the database name. If the data base name is incorrect the message "YANLIŞ DOSYA İSMİ" will be displayed. If BUILDSORGU can not find a data base with the given name it displays the message "YANLIŞ VERİ TABANI".

BUILDSORGU will respond with the message

```
SADECE SORGULAMA (EVET VEYA HAYIR)?
```

the user should enter "EVET" if record update, creation, and deletion are to be disallowed. If the response is "HAYIR" then it displays the message

```
GÜNLEME (EVET VEYA HAYIR)?
```

"EVET" allows record update; a "HAYIR" response disallows record update.

```
YARATMA (EVET VEYA HAYIR)?
```

"EVET" allows create; "HAYIR" disallows create.

```
ATMA (EVET VEYA HAYIR)?
```

"EVET" allows delete; "HAYIR" disallows delete.

BUILDSORGU will then display

HANGİ OPSİYON

1 TÜM VERİ TABANI

2 SEÇİLMİŞ VERİ GRUPLARI

3 MANTIKSAL VERİ TABANI

If there are no logical data bases in the data base then option 3 is not displayed.

Option 1 allows access to all data base structures.

Option 2 allows access to selected data base structures. BUILDSORGU will display the following

AŞAGIDAKİ VERİ GRUPLARI İÇİN

D SORGU DIŞINDA BIRAKMAK

K SORGU İÇİNE KABUL

B SEÇMELER BİTTİ

x PROGRAMI BİTİRMEK

Following this, BUILDSORGU displays the name of each disjoint data set in the data base, and waits for a response after each name displayed. Option 3 allows access to one logical data base, and displays the following

MANTIKSAL VERİ TABANI İSMİ

The user should enter the logical data base name.

BUILDSORGU then displays

SORGU PROGRAM ADI (DEFAULT BOS)?

"(DEFAULT BOS)" appears only when the options 1 or 3 is chosen and the data base name does not contian any special characters. If null input is entered the default name is OBJECT/SORGU/sorgu name is used where sorgu name is the data base name in option 1, and the logical data base name in option 3.

BUILDSORGU then displays

HANGİ QUEUE (DEFAULT BOŞ)?

If a null input is entered the compilation is done in the system default queue; otherwise in the queue specified by the user.

BUILDSORGU then displays

```
TANIMLAMA DOSYA ADI (DEFAULT BOŞ)?
```

If a null input is entered the default file name will be (USERCODE) DEFINITIONS/sorgu name.

Running SORGU using a terminal.

The user should log-on to CANDE in order to run the program SORGU. A typical log-on is as follows

```
HELLO
B6800 CANDE
ENTER USERCODE PLEASE
usercode/password
SESSION
```

After log-on is performed SORGU can be initiated by entering

```
RUN SORGU/sorgu name
```

SORGU responds

```
SORGU BAŞLIYOR
"veri tabanı adı" HAZIR
```

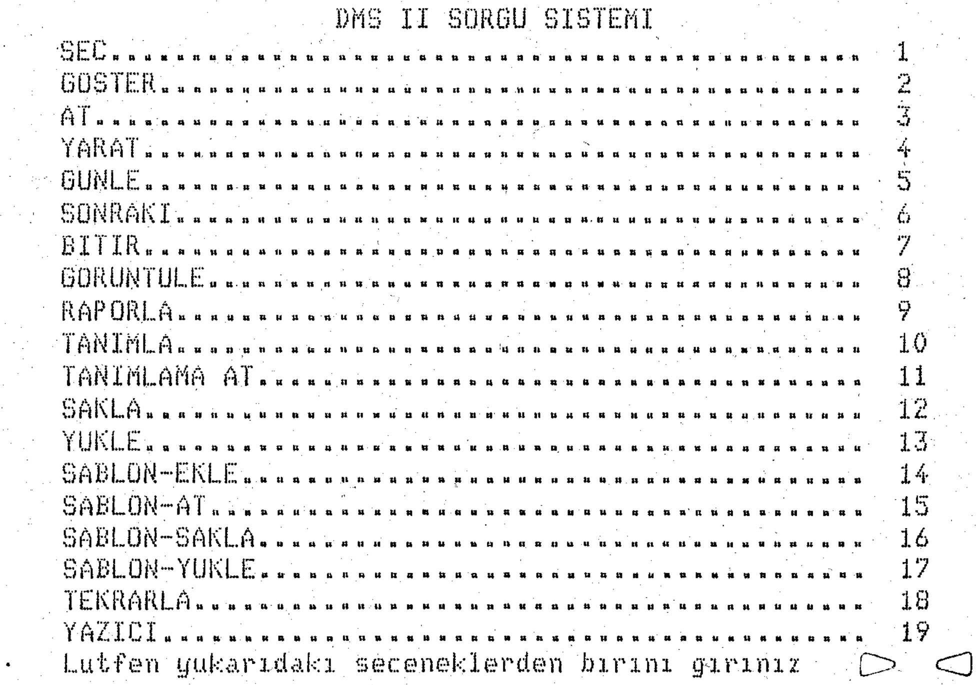
At this moment the user can enter INQUIRY statements, if he wants to branch to SORGU he should enter

```
SORGU
```

and the main menu of SORGU will be displayed. Whenever the user wishes to use the INQUIRY commands instead of SORGU he should only enter 20 in the main menu and the SORGU will act as an INQUIRY.

4.5. SORGU Operations

Whenever, a user enters the keyword "SORGU" or after completion of a SORGU operation;



```
                    DMS II SORGU SISTEMI
SEC.............................................  1
GOSTER..........................................  2
AT..............................................  3
YARAT...........................................  4
GUNLE...........................................  5
SONRAKI.........................................  6
BITIR...........................................  7
GORUNTULE.......................................  8
RAPORLA.........................................  9
TANIMLA.........................................  10
TANIMLAMA AT....................................  11
SAKLA...........................................  12
YUKLE...........................................  13
SABLON-EKLE.....................................  14
SABLON-AT.......................................  15
SABLON-SAKLA....................................  16
SABLON-YUKLE....................................  17
TEKRARLA........................................  18
YAZICI..........................................  19
Lutfen yukarıdakı seceneklerden bırını gırınız
```

Fig. 4.1 Main Menu of SORGU

appears.

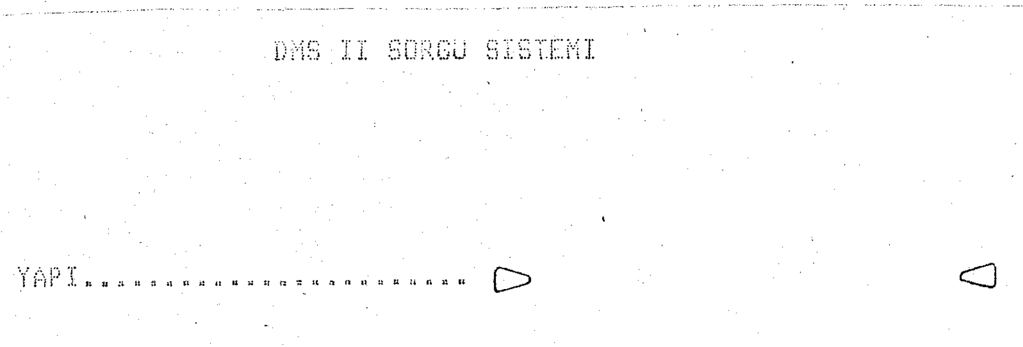
- The user can enter the related code for the operation to be done.
- Exceptional cases:
  - If the user enters "20" SORGU disappears and the functions of INQUIRY become visible.
  - If the user enters a null input or a code greater then "20" then the message "YANLIŞ KOD; Lütfen düzeltin" appears at the 23rd line of the terminal.

1. SEÇ:

Related code is 1.

Function:

Locating records which satisfy the condition(s) specified by the user. When the user enters "1" at the main menu;



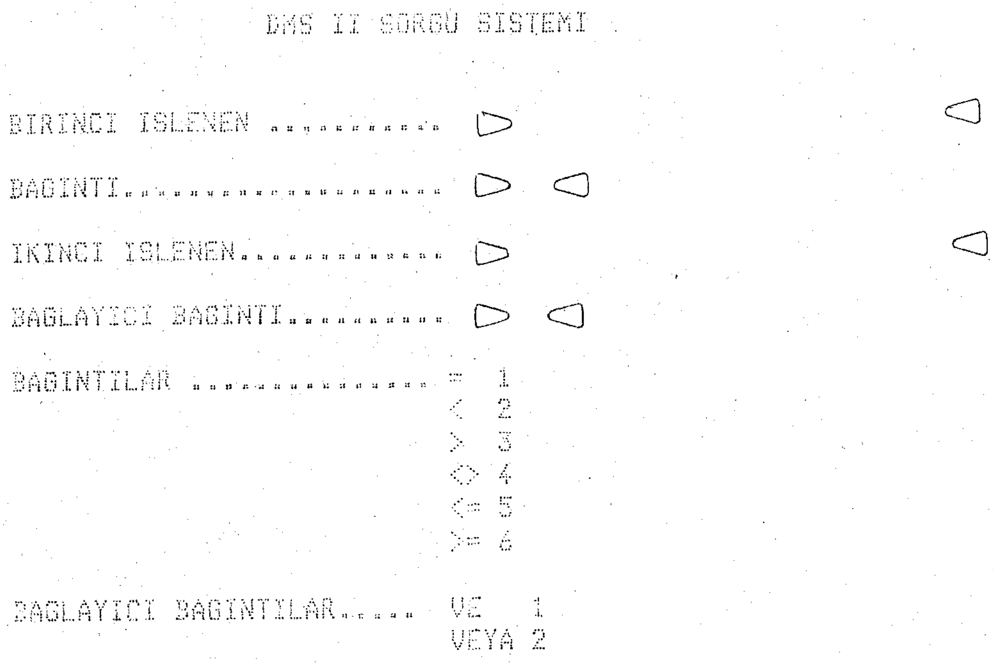
```
                DMS II SORGU SISTEMI



YAPI........................  ▷                          ◁
```

Fig. 4.2 Menu for Structure Name

Appears.

- If the user enters a null input the user is returned back to the main menu.
- Exceptional case:

  If the user uses a mapped structure name which is not valid the message "Yanlış YAPI; Lütfen düzeltin" appears at the 23rd line of the terminal. After entering a valid structure name;



```
                 DMS II SORGU SISTEMI

BIRINCI ISLENEN ............  ▷                         ◁

BAGINTI.....................  ▷  ◁

IKINCI ISLENEN..............  ▷                         ◁

BAGLAYICI BAGINTI...........  ▷  ◁

BAGINTILAR ................ =  1
                            <  2
                            >  3
                            <> 4
                            <= 5
                            >= 6

BAGLAYICI BAGINTILAR......  VE   1
                            VEYA 2
```

Fig. 4.3 Menu for Forming Conditions

Appears.

- The conditions are formed using this menu. THe user should enter the first operand into BİRİNCİ İŞLENEN, the relational operator into BAĞINTI, the second operand into İKİNCİ İŞLENEN, and if the condition formation process will be continued the connecting relational operator into BAĞLAYICI BAĞINTI.
- If the user changes his mind about the structure name and if this is the first time this menu appears the user enters a null input and returns to the previous menu.
- Exceptional cases.
  - If the user does not enter a valid mapped-item name for the first or second operand the messages "Yanlış BİRİNCİ İŞLENEN; Lütfen düzeltin" or "Yanlış İKİNCİ İŞLENEN; Lütfen düzeltin" appears at the 23rd line respectively.
  - If the user enters an ivalid relational operator then the message "Yanlış BAĞINTI; Lütfen düzeltin" appears.
  - If the user enters an ivalid connecting relational operator not ve or

veya then the message "Yanlış BAĞLAYICI BAĞINTI; Lütfen düzeltin" appears.

- If the user changes his mind after entering the first part of the condition and wishes to return to change the structure name this is forbidden. In this the message "Lütfen işlenenleri giriniz" appears at the 23rd line of the terminal.

2. GÖSTER:

Related code is 2.

Function:

Allows items of a selected record to be displayed, also like SEÇ selects records that satisfy the condition(s) and displays them without locating them. When the user enters "2" at the main menu Fig. 4.2 appears.

- If the user enters a null input all items in the previously selected record is displayed.

From this point on the same actions are taken as in command SEÇ, also the same error messages are displayed. The main difference between the commands SEÇ and GÖSTER are:

- You can not update a record when the user uses the command "GÖSTER".
- The command SEÇ does not display the values of the items in the selected record. One can display them by using the command "GÖSTER".
- SEÇ selects only one record but GÖSTER displays all the records that satisfies the conditions.

3. AT:

Related code is 3.

Function:

Causes a record to be deleted from the data base. When the user enters "3" at the main menu Fig. 4.2 appears.

- If the user enters a null input return to main menu is performed.
- If the user uses a mapped structure and if it is not valid the message "Yanlış YAPI; Lütfen düzeltin" appears at the 23rd line of the terminal. WARNING: The user should select (SEÇ) a valid record from the structure he would like to delete.

4. YARAT:

Related code is 4.

Function:

Causes a new record to be created in the data base. When the user enters "4" at the main menu Fig. 4.2 appears.

The same actions are taken as "AT" but this command needs more information for forming the query. Fig. 4.3 appears and the user should enter the values for the items.

- If mapped items are used and if they are invalid one of the following messages is displayed;
  "Yanlış BİRİNCİ İŞLENEN: Lütfen düzeltin",
  "Yanlış İKİNCİ İŞLENEN; Lütfen düzeltin",
- If the BAĞINTI is not 1 (=) then the message "Yanlış BAĞINTI; Lütfen düzeltin" appears. Because only 1 is permitted for this option.
- If the BAĞLAYICI BAĞINTI is not 1 (VE) then the message "Yanlış BAĞLAYICI BAĞINTI; Lütfen düzeltin" appears because VEYA is not valid.

5. GONLE:

Related code is 5.

Function:

Causes a record to be deleted from the data base. This command acts completely like the command YARAT.

6. SONRAKİ:

Related code is 6.

Function:

Causes SORGU to continue record selection and item display of the most recently entered command from the point at which it stopped. When the user enters "6" at the main menu Fig. 4.2 appears.

- If the user enters a structure name then the continuation is done on that structure.
- If the user does not specify any structure then the structure in the most recently processed query is continued.
- If the a mapped name is used for the structure and if it is invalid the message "Yanlış YAPI; Lütfen düzeltin" appears at the 23rd line of the terminal.

7. BİTİR:

Related code is 7.

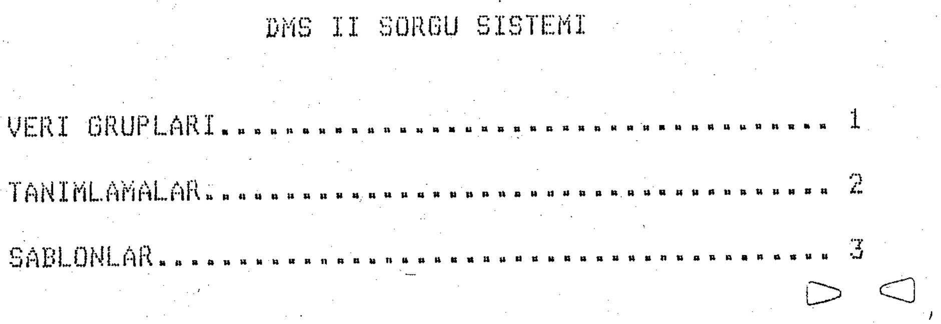
Function:

Terminates the SORGU session. The user is not asked for any other input after entering "7" to the main menu.

8. GÖRÜNTÜLE:

Related code is 8.

Function: Displays all or selected portions of the data base description and may also be used to display the defined reports and mappings. When the user enters "8" to the main menu;

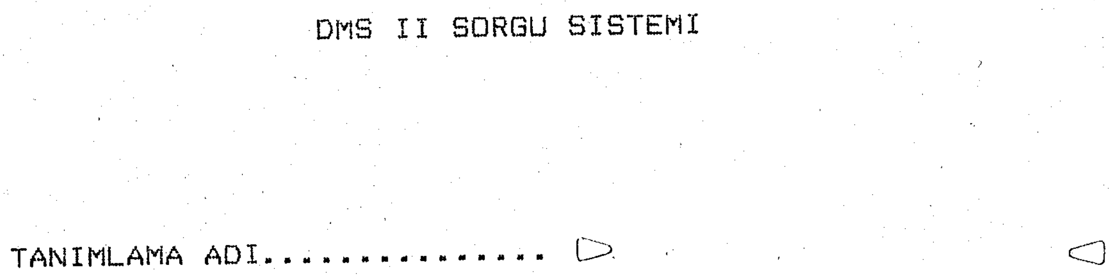


```
                  DMS II SORGU SISTEMI

VERI GRUPLARI.......................................... 1

TANIMLAMALAR........................................... 2

SABLONLAR.............................................. 3
```

Fig. 4.4 Menu for Show Options

Appears.

- If the user enters "1" to this option selection, Fig. 4.2 appears;
  - If the user enters a null input the data set names are displayed.
  - If the user enters a structure (data set name) name then the items in that data set is displayed.
  - If the user enters "2" to the option selection;



DMS II SORGU SISTEMI

TANIMLAMA ADI............... ▷ ◁

Fig. 4.5 Menu for Define-item Names

Appears.

- If the user enters a null input the names and text of all defined reports are displayed.
- If the user enters a define name then the text of the previously defined report is displayed.

- If the user enters "3" to the option selection the previously defined mapping relations are displayed. If the user wants to break this sequence he should enter "K" and return to main menu will be performed.

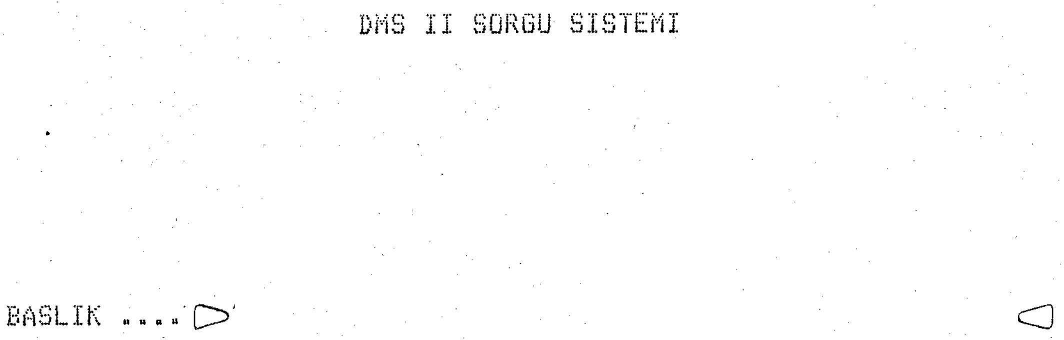
9. RAPORLA:

Related code is 9.

Function:

Controls both the type of items that are to be listed on a report as well as the format of the report. When the user enters "9" to the main menu Fig. 4.2 appears.

- If the user enters a null input return to main menu is performed.
- If the user enters a structure name the report will be generated using that structure. When user enters the structure name;



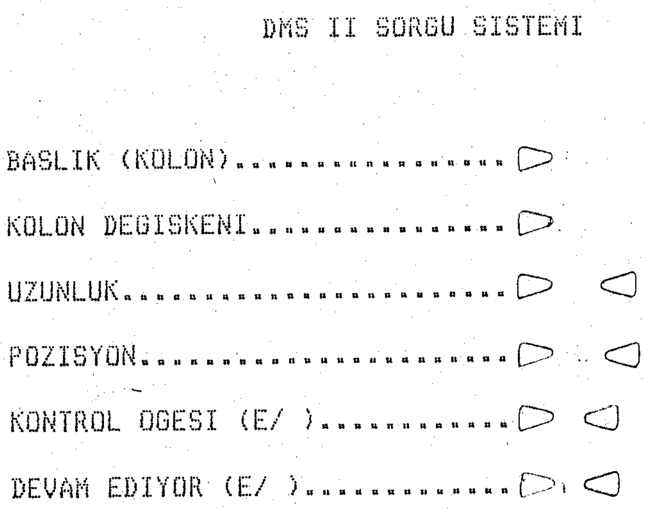
```
                      DMS II SORGU SISTEMI




BASLIK ....▷                                                      ◁
```

Fig. 4.6 Menu for Report Headings

Appears.

- The user should enter the page heading of the report; if he wishes not to have a page heading then he should enter a null input.

After inputting the heading;



```
                DMS II SORGU SISTEMI


BASLIK (KOLON)................. ▷                        ◁

KOLON DEGISKENI................ ▷                        ◁

UZUNLUK........................ ▷  ◁

POZISYON....................... ▷  ◁

KONTROL OGESI (E/ )............ ▷ ◁

DEVAM EDIYOR (E/ )............. ▷ ◁
```

Fig. 4.7 Menu for Report Items

Appears.

- The elements of the report are determined through this menu. The user should enter the column heading into the "BAŞLIK (KOLON)", the item that will be used on the given column into the "KOLON DEĞİŞKENİ", the length of the item into the "UZUNLUK", the position (column) of the item into the "POZİSYON", if it is a controlling element then the user should enter "E" into the "KONTROL ÖĞESİ", and if this item is not the last item in the report the user should enter "E" into the "DEVAM EDİYOR".
- If the length of the item defined in the DASDL source is greater then the value in UZUNLUK, then the input is ignored.
- While inputting the items in the report whenever the user determines whether the item is controlling or not he should be careful. The controlling items should be entered prior to the other items.
- The column heading can not be more than one word it should be a single string. Also the special Turkish characters can not be used.

- If the user enters null input then a return to the previous menu is performed.
- If the user does not enter item name then the message "KOLON DEĞİŞKENİ gerekli; Lütfen düzeltin" appears at the 23rd line of the terminal.
- If the user enters an input other than null or "E" to the KONTROL ÖĞESİ then the message "KONTROL ÖĞESİ (E/) olabilir; Lütfen düzeltin" appears.
- If the user enters an input other than null or "E" to the DEVAM EDİYOR then the message "DEVAM KODU (E/) olabilir; Lütfen düzeltin" appears at the 23rd line of the terminal.
- After inputting a non-controlling item the user can not enter a controlling item, if he tries to enter the message "KONTORL ÖĞESİ girilemez; Lütfen düzeltin" appears at the 23rd line.
- When the user enters a null input into the DEVAM EDİYOR then Fig. 4.2 appears, the user should enter the name of the data set to be used. After entering the data set name the query is ready and the report appears after three " " signs.

10, TANIMLA:

Related code is 10.

Function allows the report definition text to be assigned a name. When SQRGU sees this define name, it replaces the define name with the associated text. When the user enters "10" into the main menu Fig. 4.5 appears. The user should enter the name of the report to be defined. This command acts completely the same as the command RAPORLA from this point on. The only difference between these two is that the query is executed in the command RAPORLA but using this command only the definition of report is prepared. In order to execute a report defined using this command option "20" should be elected at the main menu.

11. TANIMLA AT:

Related code is 11.

Function:

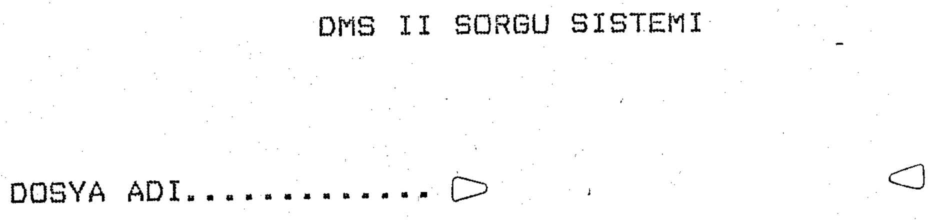
Discards previously defined reports using the command TANIMLA. When the user enters "11" into the main menu Fig. 4.5 appears. The user should enter the name of the previously defined report. After inputting the define name the query is ready and the report will be discarded.

12. SAKLA:

Related code is 12.

Function:

Stores the text of defined items in a file on disk. These texts can be reloaded and used during subsequent SORGU sessions. When the user enters "12" at the main menu;



```
                    DMS II SORGU SISTEMI


DOSYA ADI............. ▷                         ◁
```

Fig. 4.8 Menu for File Name

Appears.

- The user should enter the name of the file where report definitions will be saved. If the user enters a null input then the message "DOSYA ADI gerekli; Lütfen giriniz" appears at the 23rd line of the terminal.

13. YÜKLE:

Related code is 13.

Function:

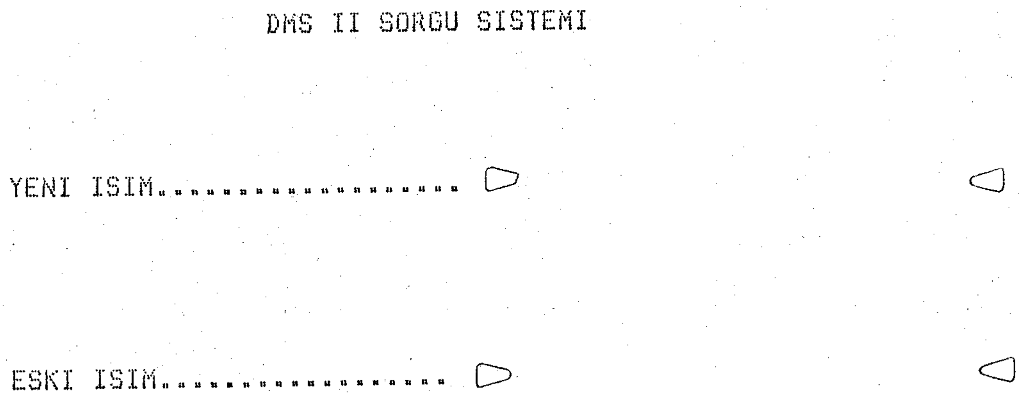
Allows previously saved text to be retrieved. When the user enters "13" into the main menu Fig. 4.8 appears. This command is similar to the previous command SAKLA. But this command loads the previously defined texts in other words this the inverse function of the previous command.

14. ŞABLON EKLE:

Related code is 14.

Function:

Defining a mapping relation between an item name in the data base and a user defined name. When the user enters "14" to the main menu.



```
                    DMS II SORGU SISTEMI


YENI ISIM.................... ▷                        ◁



ESKI ISIM.................... ▷                        ◁
```

Fig. 4.9 Menu for Mapping Relations

Appears.

- The user should enter the user defined name for the item into the YENİ İSİM and the old text into the ESKİ İSİM.
- The user is restricted to 100 mapping relations in a session of an attempt to define the 101st relation is made the message "100 den

fazla ŞABLON EKLEnemez" appears at the 23rd line and a return to the main menu is performed.

- If the user tries to enter a previously defined name the system responds with the message "Aynı isim daha önce ŞABLON EKLEnmiş; Lütfen düzeltin".

15. ŞABLON AT:

Related code is 15.

Function:

Deleting a mapping relation from the list of mapping relations. When the user enters "15" to the main menu Fig. 4.9 appears. The user should enter both the user defined name and the original text in order to delete from the list.

- If the system does not find any match the message "Aynı isim daha önce ŞABLON EKLEnmemiş; Lütfen düzeltin" appears.

16. ŞABLON SAKLA:

Related code is 16.

Function:

Saving the mapping relations in a file named "MAPS". When the user enters "16" to the main menu the query is performed automatically.

17. ŞABLON YÜKLE:

Related code is 17.

Function:

Loading previously defined mapping relations from the MAPS file. When the user enters "17" to the main menu the query is complete.

18. TEKRARLA:

Related code is 18.

Function:

Causes re-execution of a previous SORGU command. When the user enters "18" to the main menu Fig. 4.2 appears.

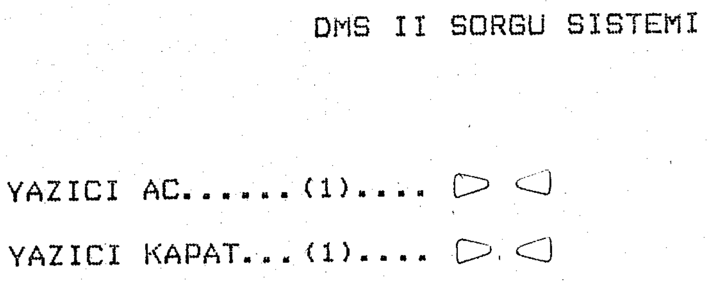
- If the user enters a structure name then the last command on that structure is executed.
- If the user does not specify any structure then the last command is re-executed.
- If a mapped name is used for the structure and if it is invalid then the message "Yanlış YAPI; Lütfen düzeltin" appears at the 23rd line of the terminal.

19. YAZICI:

Related code is 19.

Function:

Setting or resetting the printer option. When the printer option is set then the results of queries are written into hardcopy and not displayed. When the user enters "19" at the main menu;



```
                    DMS II SORGU SISTEMI


YAZICI AC......(1).... ▷ ◁
YAZICI KAPAT...(1).... ▷ ◁
```

Fig. 4.10 Menu for printer options

Appears.

- If the user wants to set the printer option he should enter "1" into the YAZICI AÇ.
- If the user wants to reset the printer option he should enter "1" into the YAZICI KAPAT.
- If the user enters a null input or "1" to both the YAZICI AÇ and

YAZICI KAPAT or an input other then "0" or "1" then the message "Yanlış KOD; Lütfen düzeltin" appears at the 23rd line of the terminal.

V. CONCLUSION

The most interesting and useful feature of SORGU is the user defined report. In SORGU these reports can be produced with the use of 4 or 5 menus, whereas using BDMSCOBOL the same report can only be generated by writ.ing almost 100 statements. One important restriction of SORGU is that one can only produce reports dealing with one and only one data set. This restriction is not created by SORGU, it exitsts in INQUIRY also. When this problem is solved in INQUIRY adaptation in SORGU is a simple task. The existing menu-driven nature of SORGU can be extended to handle reports with multiple sets.

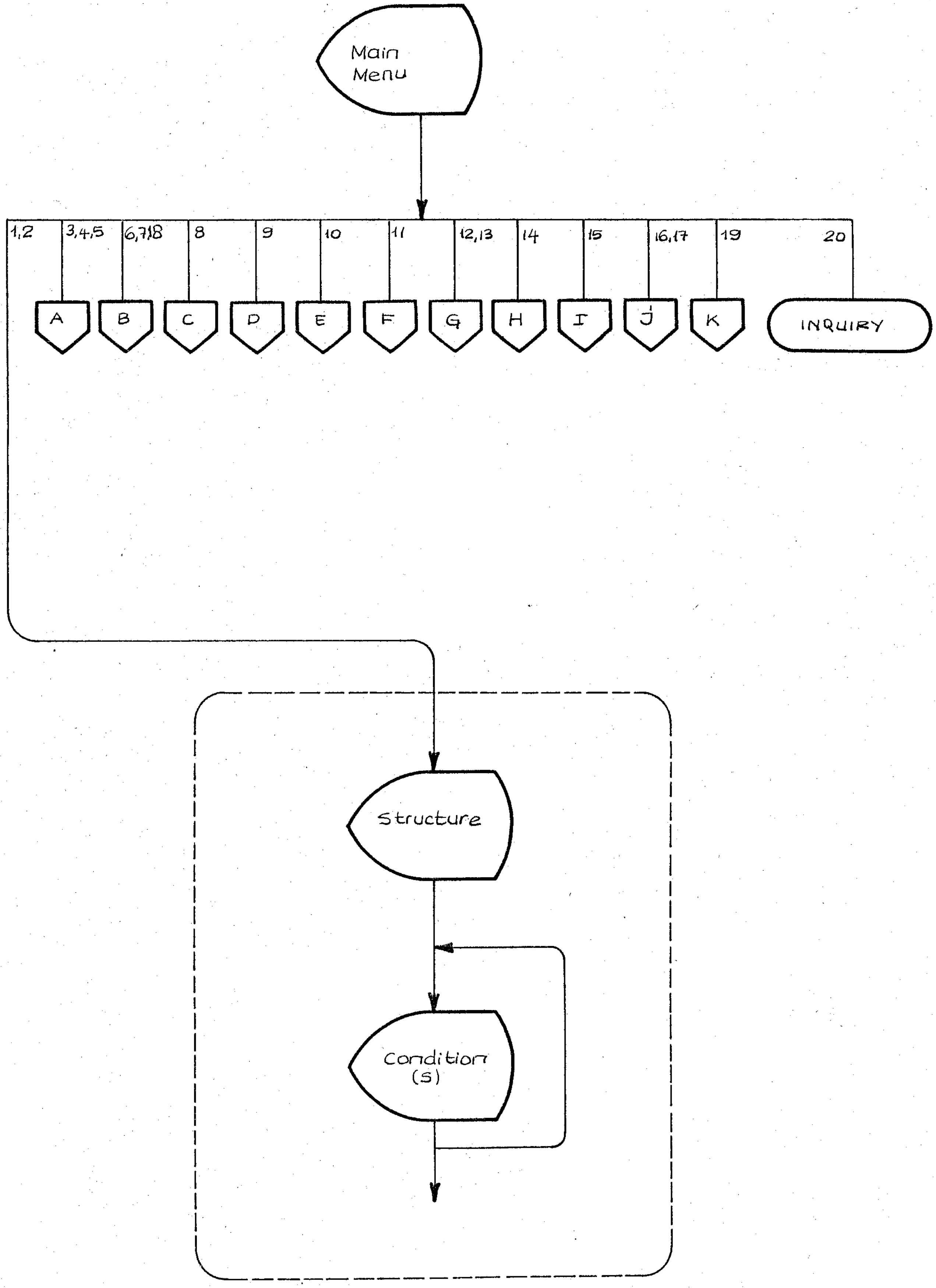
Until now SORGU has not been used formally. It is expected that SORGU will be used instead of INQUIRY at Arçelik A.Ş. While testing the software several prospective users were interested in SORGU and they have used SORGU for a short period of time. SORGU can easily be installed on the data bases developed at Arçelik A.Ş. The example in chapter 3 is based on the data base developed for Inventory Control and Purchasing of Raw Materials. SORGU facilitates the easy generation of queries and reports. As such it can be used either by non-EDP management personnel for assistance in reaching decisions or by EDP personnel to generate the reports requested by management. It was hoped that the management would be able to obtain the required reports directly and immidiately through the use SORGU. However, one should not expect widespread use of SORGU among management personnel at Arçelik A.Ş. since most of them would rather delegate this job to the EDP personnel. The current version of SORGU can be installed in every data base developed for the Burroughs large systems using DMS-II data base management system.

There are a lot of users that use interactive information retrieval software packages. SORGU provides a convenient but restrictive environment to Turkish users. However, the translation of SORGU messages into other languages is a relatively simple exercise. The ultimate step has to be the development of a completely natural language-like inquiry software. The specification of queries with many parameters tends to be awkward when not supported by forms or formatted display screens (menus). It seems then much more desirable to allow free-form natural language-like input. This can also provide the ability to have expressions of much greater complexity than can be obtained using formatted queries. It can not be expected now that computers will understand English or any other language natural to human beings. The problem of using natural language in queries in data base environment is much less than the problem of understanding natural language in general. In a data base the user will operate in a well-defined context. A limited area of discourse makes it possible to avoid many of the ambiguities of natural languages. The translation of natural language-like inquiry software is almost impossible. Because the natural languages do differ a lot. For example there are a lot of words and phrases in Turkish that mean the same as "or" in English. Therefore this kind of software is not easily transportable to other natural languages.

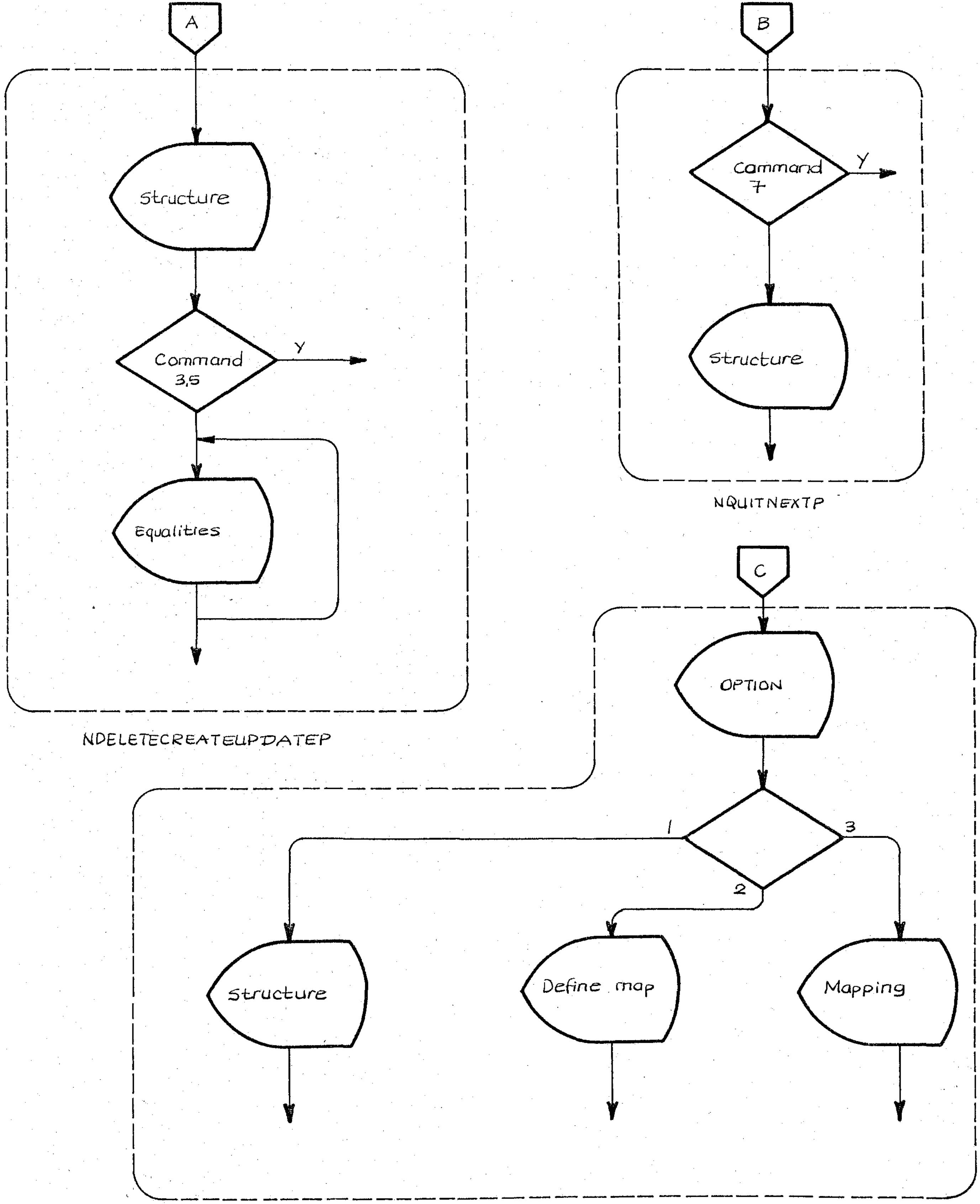
Much work remains to be done in the area of query processing. It is necessary to understand the needs of the user and the human factors in information search in order to find the proper balance between flexibility and expressiveness, ease of use and tedium.

APPENDICES

APPENDIX A.MENU BRANCHING IN SORGU



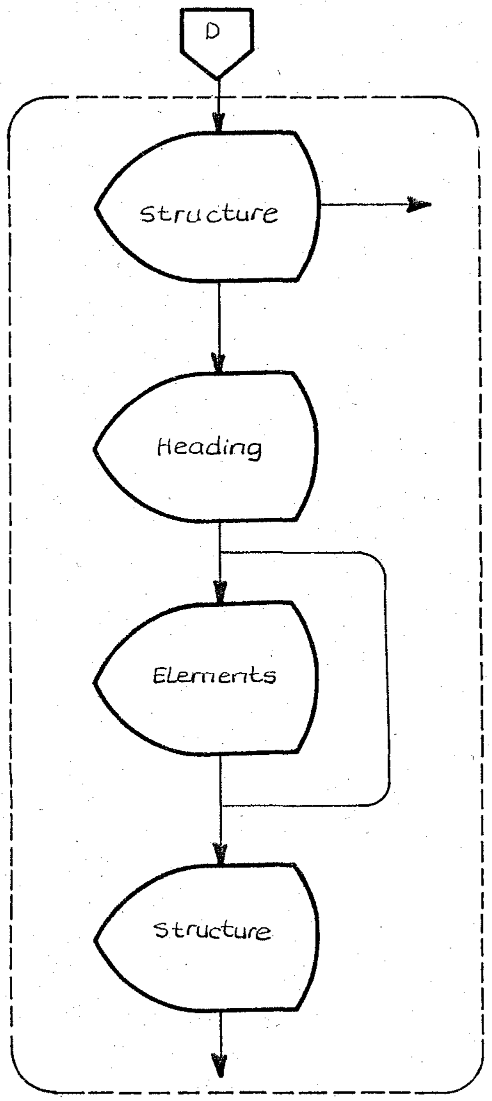
N SELECTDISPLAYP



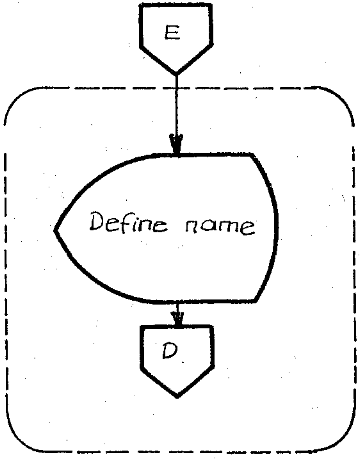
NDELETECREATEUPDATEP

NQUITNEXTP

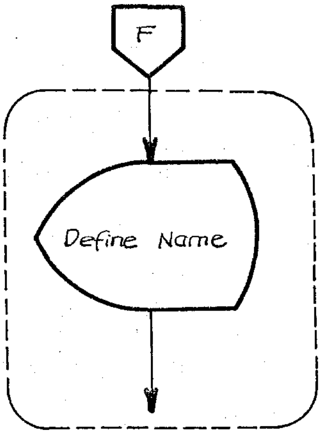
NSHOWP



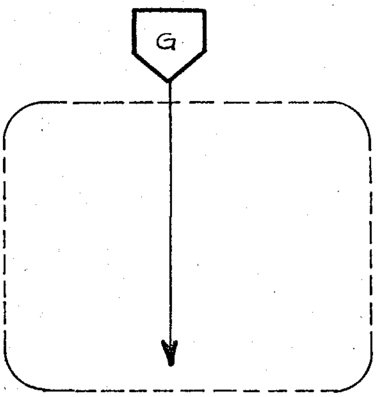
NREPORTP



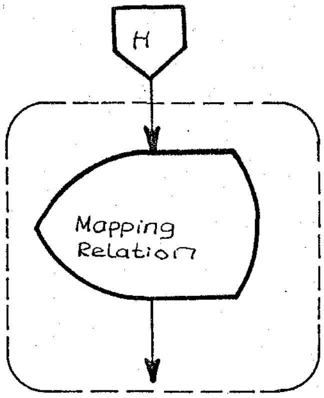
NDEFINEP



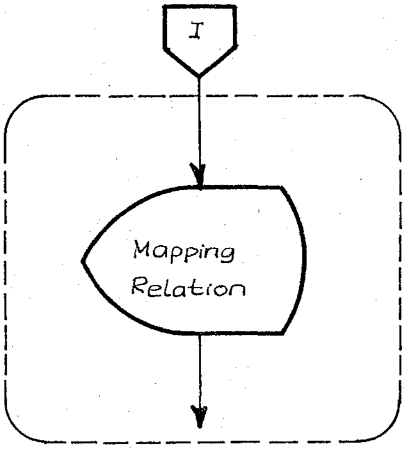
NCLEARP



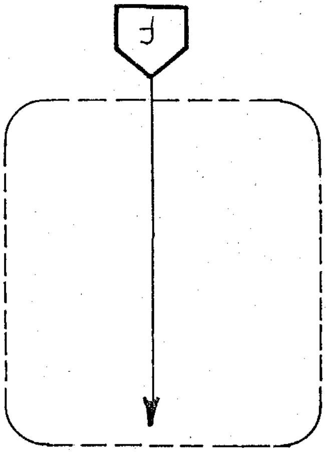
NSAVERESTOREP



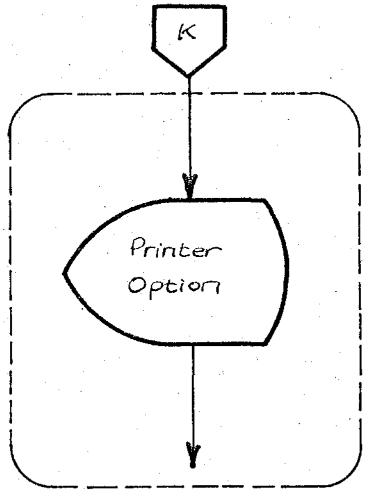
NTANIMLAP



NTANIMLAATP



NMAPSAVERESTOREP



APPENDIX B. THE RELATION BETWEEN THE COMMANDS OF SORGU AND INQUIRY.

SEÇ	SELECT
GÖSTER	DISPLAY
AT	DELETE
YARAT	CREATE
GÜNLE	UPDATE
SONRAKİ	NEXT
BİTİR	QUIT
GÖRÜNTÜLE	SHOW
RAPORLA	REPORT
TANIMLA	DEFINE
TANIMLAMA AT	CLEAR
SAKLA	SAVE
YÜKLE	RESTORE
ŞABLON EKLE	
ŞABLON AT	
ŞABLON SAKLA	
ŞABLON YÜKLE	
TEKRARLA	REPEAT
YAZICI	OPTION PRINTER

APPENDIX C.SYSTEM GENERATION

The information given below is valid for Burroughs large systems (B5000/B6000/B7000/A9 Series of Systems), for other Burroughs systems SORGU is not available because it is written in DMALGOL and this language is not supported.

SORGU files:

DMINQ/BUILDSYMBOLIC

Symbolic of the program which builds INQUIRY.

DMINQ/BUILDSYMBOLIC/PATCHES/FOR/SORGU

Symbolic patches for the program BUILDSORGU.

OBJECT/BUILDSORGU

Object code for the program that prepares SORGU or INQUIRY.

DMINQ/SYMBOLIC

Symbolic for INQUIRY.

DMINQ/SYMBOLIC/PATCHES/FOR/SORGU

Symbolic patches for extensions that creates SORGU from INQUIRY.

OBJECT/SORGU

Object code for the program SORGU.

DMINQ/HELP

Contains the information returned to user in response to the HELP statement. (Not valid for SORGU)

Generation of BUILDSORGU.

The following cards must be included in a job deck and executed.

```
?COMPILE OBJECT/BUILDSORGU WITH DMALGOL LIBRARY
?COMPILER FILE TAPE = DMINQ/BUILDSYMBOLIC
?COMPILER FILE CARD (KIND=DISK, TITLE=DMINQ/BUILDSYMBOLIC/PATCHES/FOR/SORGU)
```

After running the program BUILDSORGU the definition of the SORGU program has been completed. The user then should prepare the following cards, which must be included in a job deck and executed.

```
?COMPILE OBJECT/SORGUsorgu name WITH DMALGOL LIBRARY
?COMPILER FILE TAPE = DMINQ/SYMBOLIC
?COMPILER FILE CARD (KIND-DISK, TITLE=DMINQ/SYMBOLIC/PATCHES/FOR/SORGU)
?COMPILER FILE DASDL - DMINQDIRECTORY/sorgu name
?COMPILER FILE PROPERTIES = DATABASE/PROPERTIES
```

After executing this card deck SORGU software is ready.

APPENDIX D.LISTING OF MODIFICATIONS TO SORGU

```
$MERGE                                                                  00000001
         MAXRECSIZE = 1927);                                            10069000
   FILE DISKE(KIND=DISK,MAXRECSIZE=080,BLOCKSIZE=800,MYUSE=OUT,UNITS=1, 10073010
                  AREASIZE=8000,TITLE="DISKIM." );                      10073015
   FILE DISKTEN(KIND=DISK,MAXRECSIZE=080,BLOCKSIZE=800,MYUSE=IN,UNITS=1,10073020
                  AREASIZE=8000,TITLE="DISKIM." );                      10073025
   FILE MAPSAVEW(KIND=DISK,MAXRECSIZE=60,BLOCKSIZE=180,UNITS=1,MYUSE=OUT10073030
                  ,TITLE="MAPS.");                                      10073035
   FILE MAPSAVER(KIND=DISK,MAXRECSIZE=60,BLOCKSIZE=180,UNITS=1,MYUSE=IN,10073040
                   TITLE="MAPS.");                                      10073045
   EBCDIC ARRAY BUF[0:1926];                                            21000000
INTEGER KAYITSAYISI;                                                    21000005
INTEGER I;                                                              21000010
INTEGER GERIYEDON;                                                      21000020
STRING BIRINCIOP;                                                       21000025
STRING IKINCIOP;                                                        21000030
STRING YAPI;                                                            21000035
STRING NFIRSTOP;                                                        21000040
STRING YAZ;                                                             21000045
INTEGER NFOPERATOR;                                                     21000050
STRING NSECONDOP;                                                       21000055
STRING ARRAY NU [1:100,1:2];                                            21000060
INTEGER NUPOINTER;                                                      21000065
INTEGER NSOPERATOR;                                                     21000070
INTEGER KOMUT;                                                          21000075
INTEGER NCONT;                                                          21000080
INTEGER NCONT2;                                                         21000082
FORMAT FMT (A080);                                                      21000085
   POINTER PBUF;                                                        21000090
   DEFINE SAY = REPLACE PBUF:PBUF BY #                                  21000095
   ;                                                                    21000100
   DEFINE TAK = REPLACE LAF:LAF BY #                                    21000105
   ;                                                                    21000110
   POINTER LAF    ;                                                     21000115
   LIBRARY NL(                                                          29000000
             FUNCTIONNAME = "TD830SUPPORT."                             29000005
            ,LIBACCESS = BYFUNCTION);                                   29000010
   PROCEDURE CURSOR(ROW,COL,Q);                                         29000015
             VALUE ROW,COL;                                             29000020
             INTEGER ROW,COL;                                           29000025
             POINTER Q;                                                 29000030
             LIBRARY NL;                                                29000035
   PROCEDURE WRYTE(FYLE,P);                                             29000040
             FILE FYLE;                                                 29000045
             POINTER P;                                                 29000050
             LIBRARY NL;                                                29000055
$PAGE                                                                   29000120
   PROCEDURE NFORMS(NO);VALUE NO;INTEGER NO;                            29000125
   BEGIN                                                                29000130
      CASE NO OF                                                        29000135
      BEGIN                                                             29000140
$PAGE                                                                   29000145
        0:                                                              29000150
   SAY 48"0C";                                                          29000155
      CURSOR(2,24,PBUF);                                                29000160
      SAY " DMS II SORGU SISTEMI";                                      29000165
      CURSOR(3,8,PBUF);                                                 29000170
      SAY "SEC.....................................";                   29000175
      SAY "..........  1";                                              29000180
      CURSOR(4,8,PBUF);                                                 29000185
      SAY "GOSTER.................................";
      SAY "..........  2";
      CURSOR(5,8,PBUF);
      SAY "AT.....................................";
      SAY "..........  3";
      CURSOR(6,8,PBUF);
      SAY "YARAT..................................";
      SAY "..........  4";
      CURSOR(7,8,PBUF);
      SAY "GUNLE..................................";
      SAY "..........  5";
      CURSOR(8,8,PBUF);
      SAY "SONRAKI................................";
      SAY "..........  6";
      CURSOR(09,8,PBUF);
      SAY "BITIR..................................";
      SAY "..........  7";
      CURSOR(10,8,PBUF);
      SAY "GORUNTULE..............................";
      SAY "..........  8";
      CURSOR(11,8,PBUF);
      SAY "RAPORLA................................";
      SAY "..........  9";
      CURSOR(12,8,PBUF);
      SAY "TANIMLA................................";
      SAY "..........  10";
      CURSOR(13,8,PBUF);
      SAY "TANIMLAMA AT...........................";
      SAY "..........  11";
      CURSOR(14,8,PBUF);
      SAY "SAKLA..................................";
      SAY "..........  12";
      CURSOR(15,8,PBUF);
      SAY "YUKLE..................................";
      SAY "..........  13";
      CURSOR(16,8,PBUF);
      SAY "SABLON-EKLE............................";
      SAY "..........  14";
      CURSOR(17,8,PBUF);
      SAY "SABLON-AT..............................";
      SAY "..........  15";
      CURSOR(18,8,PBUF);
      SAY "SABLON-SAKLA...........................";
      SAY "..........  16";
      CURSOR(19,8,PBUF);
      SAY "SABLON-YUKLE...........................";
      SAY "..........  17";
      CURSOR(20,8,PBUF);
      SAY "TEKRARLA...............................";
      SAY "..........  18";
      CURSOR(21,8,PBUF);
      SAY "YAZICI.................................";
      SAY "..........  19";
      CURSOR(22,8,PBUF);
      SAY "Lutfen yukaridaki seceneklerden birini g";
      SAY "iriniz          ";
      SAY 48"27E603";
      WRYTE(REMOTE,PBUF);
$PAGE
        1:
```

```
   SAY 48"OC";                                                          29000490
      CURSOR(3,24,PBUF);                                                29000495
      SAY "DMS II SORGU SISTEMI";                                       29000500
      CURSOR(10,8,PBUF);                                                29000505
      SAY "YAPI.......................   ";                             29000510
      CURSOR(10,56,PBUF);                                               29000515
      SAY " ";                                                          29000520
      SAY 48"27E603";                                                   29000525
      WRYTE(REMOTE,PBUF);                                               29000530
$PAGE                                                                   29000535
        2:                                                              29000540
   SAY 48"OC";                                                          29000545
      CURSOR(3,24,PBUF);                                                29000550
      SAY "DMS II SORGU SISTEMI";                                       29000555
      CURSOR(6,8,PBUF);                                                 29000560
      SAY "BIRINCI ISLENEN ...........   ";                             29000565
      CURSOR(6,62,PBUF);                                                29000570
      SAY " ";                                                          29000575
      CURSOR(8,8,PBUF);                                                 29000580
      SAY "BAGINTI....................     ";                           29000585
      CURSOR(10,8,PBUF);                                                29000590
      SAY "IKINCI ISLENEN.............   ";                             29000595
      CURSOR(10,62,PBUF);                                               29000600
      SAY " ";                                                          29000605
      CURSOR(12,8,PBUF);                                                29000610
      SAY "BAGLAYICI BAGINTI..........     ";                           29000615
      CURSOR(14,8,PBUF);                                                29000620
      SAY "BAGINTILAR ............... =  1";                            29000625
      CURSOR(15,35,PBUF);                                               29000630
      SAY "<  2";                                                       29000635
      CURSOR(16,35,PBUF);                                               29000640
      SAY ">  3";                                                       29000645
      CURSOR(17,35,PBUF);                                               29000650
      SAY "<> 4";                                                       29000655
      CURSOR(18,35,PBUF);                                               29000660
      SAY "<= 5";                                                       29000665
      CURSOR(19,35,PBUF);                                               29000670
      SAY ">= 6";                                                       29000675
      CURSOR(21,8,PBUF);                                                29000680
      SAY "BAGLAYICI BAGINTILAR.....  VE   1";                          29000685
      CURSOR(22,35,PBUF);                                               29000690
      SAY "VEYA 2";                                                     29000695
      SAY 46"27E603";                                                   29000700
      WRYTE(REMOTE,PBUF);                                               29000705
$PAGE                                                                   29000710
        3:                                                              29000715
   SAY 48"OC";                                                          29000720
      CURSOR(3,24,PBUF);                                                29000725
      SAY "DMS II SORGU SISTEMI";                                       29000730
      CURSOR(12,2,PBUF);                                                29000735
      SAY "BASLIK ....   ";                                             29000740
      CURSOR(12,56,PBUF);                                               29000745
      SAY " ";                                                          29000750
      SAY 48"27E603";                                                   29000755
      WRYTE(REMOTE,PBUF);                                               29000760
$PAGE                                                                   29000765
        4:                                                              29000770
   SAY 48"OC";                                                          29000775
      CURSOR(3,24,PBUF);                                                29000780
      SAY "DMS II SORGU SISTEMI";                                       29000785
```

```
         CURSOR(7,8,PBUF);                                                                    29000790
         SAY "BASLIK (KOLON)..................    ";                                          29000795
         CURSOR(7,72,PBUF);                                                                   29000800
         SAY " ";                                                                             29000805
         CURSOR(9,8,PBUF);                                                                    29000810
         SAY "KOLON DEGISKENI.................    ";                                          29000815
         CURSOR(9,59,PBUF);                                                                   29000820
         SAY " ";                                                                             29000825
         CURSOR(11,8,PBUF);                                                                   29000830
         SAY "UZUNLUK........................        ";                                       29000835
         CURSOR(13,8,PBUF);                                                                   29000840
         SAY "POZISYON.......................       ";                                        29000845
         CURSOR(15,8,PBUF);                                                                   29000850
         SAY "KONTROL OGESI (E/ )............      ";                                         29000855
         CURSOR(17,8,PBUF);                                                                   29000860
         SAY "DEVAM EDIYOR (E/ ).............      ";                                         29000865
         SAY 48"27E603";                                                                      29000870
         WRYTE(REMOTE,PBUF);                                                                  29000875
$PAGE                                                                                         29000880
          5:                                                                                  29000885
     SAY 48"0C";                                                                              29000890
         CURSOR(3,24,PBUF);                                                                   29000895
         SAY "DMS II SORGU SISTEMI";                                                          29000900
         CURSOR(8,8,PBUF);                                                                    29000905
         SAY "YENI ISIM...................    ";                                              29000910
         CURSOR(8,70,PBUF);                                                                   29000915
         SAY " ";                                                                             29000920
         CURSOR(14,8,PBUF);                                                                   29000925
         SAY "ESKI ISIM..................    ";                                               29000930
         CURSOR(14,69,PBUF);                                                                  29000935
         SAY " ";                                                                             29000940
         SAY 48"27E603";                                                                      29000945
         WRYTE(REMOTE,PBUF);                                                                  29000950
$PAGE                                                                                         29000955
          6:                                                                                  29000960
     SAY 48"0C";                                                                              29000965
         CURSOR(3,24,PBUF);                                                                   29000970
         SAY "DMS II SORGU SISTEMI";                                                          29000975
         CURSOR(8,8,PBUF);                                                                    29000980
         SAY "DOSYA ADI.............    ";                                                    29000985
         CURSOR(8,50,PBUF);                                                                   29000990
         SAY " ";                                                                             29000995
         SAY 48"27E603";                                                                      29001000
         WRYTE(REMOTE,PBUF);                                                                  29001005
$PAGE                                                                                         29001010
          7:                                                                                  29001015
     SAY 48"0C";                                                                              29001020
         CURSOR(3,24,PBUF);                                                                   29001025
         SAY "DMS II SORGU SISTEMI";                                                          29001030
         CURSOR(10,8,PBUF);                                                                   29001035
         SAY "TANIMLAMA ADI...............    ";                                              29001040
         CURSOR(10,56,PBUF);                                                                  29001045
         SAY " ";                                                                             29001050
         SAY 48"27E603";                                                                      29001055
         WRYTE(REMOTE,PBUF);                                                                  29001060
$PAGE                                                                                         29001065
          8:                                                                                  29001070
     SAY 48"0C";                                                                              29001075
         CURSOR(3,24,PBUF);                                                                   29001080
         SAY "DMS II SORGU SISTEMI";                                                          29001085
```

```
      CURSOR(6,8,PBUF);
      SAY "VERI GRUPLARI............................";
      SAY "........... 1";
      CURSOR(8,8,PBUF);
      SAY "TANIMLAMALAR.............................";
      SAY "........... 2";
      CURSOR(10,8,PBUF);
      SAY "SABLONLAR................................";
      SAY "........... 3";
      CURSOR(12,58,PBUF);
      SAY "    ";
      SAY 48"27E603";
      WRYTE(REMOTE,PBUF);
$PAGE
        9:
   SAY 48"OC";
      CURSOR(3,24,PBUF);
      SAY "DMS II SORGU SISTEMI";
      CURSOR(8,8,PBUF);
      SAY "YAZICI AC......(1)....      ";
      CURSOR(10,08,PBUF);
      SAY "YAZICI KAPAT...(1)....      ";
      SAY 48"27E603";
      WRYTE(REMOTE,PBUF);
      ELSE: SAY "UNKNOWN FORMAT ";
      WRYTE(REMOTE,PBUF);
   END CASE;
END NFORMS;
$PAGE
PROCEDURE NYAZARTIK;
BEGIN
    KAYITSAYISI:=KAYITSAYISI + 1;
    WRITE (DISKE,FMT,YAZ);
END;
$PAGE
PROCEDURE ARA (PARAM,SONUC,TIP);
VALUE TIP;
STRING PARAM;
INTEGER SONUC;
INTEGER TIP;
BEGIN
LABEL NPRSONU;
LABEL CHECK;
INTEGER I;
STRING BIR,IKI;
I:=1;
SONUC:=0;
IKI:= DROP (PARAM,1);
IF TIP=1 THEN
    BIR:=TAKE (IKI,16)
ELSE
    BIR:=TAKE (IKI,23);
DISPLAY (BIR);
CHECK:
IF TIP=1 THEN
   IKI:=TAKE(NU[I,1],16)
ELSE
   IKI:=TAKE(NU[I,1],23);
DISPLAY (IKI);
IF BIR = IKI THEN
```

```
   BEGIN
       IF TIP = 1 THEN
          PARAM := TAKE( NU[I,2],16)
       ELSE
          PARAM := TAKE(NU[I,2],23);
       SONUC:=1;
       DISPLAY ("TRUE");
       GO TO NPRSONU;
   END;
I:=I+1;
IF I < NUPOINTER + 1 THEN
   GO CHECK;
NPRSONU:;
DISPLAY ("BITTI");
END;
$PAGE
PROCEDURE NSELECTDISPLAYP;
BEGIN
STRING SAKLA;
INTEGER DONUS;
LABEL A;
LABEL A1;
LABEL A01;
LABEL A0;
LABEL A2;
LABEL NSDPSONU;
INTEGER OK1;
A0:
NFORMS (1);
YAZ:=" ";
A01:
READ(REMOTE,<A17>,YAPI);
IF YAPI = "                 " THEN
   BEGIN
        IF KOMUT=2 THEN
           BEGIN
                YAZ := "D ALL";
                GO TO A2;
           END
        ELSE
           BEGIN
                BERIYEDON:=1;
                GO TO NSDPSONU;
           END;
   END;
CASE KOMUT OF
BEGIN
           1:
           YAZ:= "S ";
           2:
           YAZ:= "D ";
           ELSE :
                      BEGIN
                      SAY "DURUM KELEK";
                      WRYTE (REMOTE,PBUF)
                      END
END CASE;
IF TAKE(YAPI,1) = "$" THEN
   BEGIN
       ARA(YAPI,DONUS,1);
```

```
        IF DONUS=0   THEN
        BEGIN
           SAY 48"27E3";
           CURSOR(23,2,PBUF);
           SAY " Yanlis YAPI ;Lutfen duzeltin";
           SAY 48"27E603";
           WRYTE (REMOTE,PBUF);
           GO TO A01;
        END;
    END;
YAZ:=YAZ CAT YAPI;
YAZ:=YAZ  CAT " @";
SAKLA:=YAZ;
OK1:=0;
A:
NFORMS(2);
A1:
READ(REMOTE,(A24,I1,A24,I1),NFIRSTOP,NFOPERATOR,NSECONDOP,NSOPERATOR);
IF NFIRSTOP = "                        " OR
   NSECONDOP = "                        " THEN
         BEGIN
             IF OK1 = 0 THEN
                GO TO A0
             ELSE
                BEGIN
                SAY 48"27E3";
                CURSOR (23,2,PBUF);
                SAY " Lutfen islenenleri giriniz";
                SAY 48"27E603";
                WRYTE (REMOTE,PBUF);
                 GO TO A1;
                END
         END;
OK1:=1;
YAZ:=YAZ.CAT " ";
IF TAKE(NFIRSTOP,1) = "$" THEN
    BEGIN
        ARA(NFIRSTOP,DONUS,2);
        IF DONUS=0   THEN
        BEGIN
           SAY 48"27E3";
           CURSOR(23,2,PBUF);
           SAY " Yanlis BIRINCI ISLENEN ;Lutfen duzeltin";
           SAY 48"27E603";
           WRYTE (REMOTE,PBUF);
           YAZ:=SAKLA;
           GO TO A1;
        END;
    END;
YAZ:=YAZ CAT NFIRSTOP;
YAZ:=YAZ CAT " ";
CASE NFOPERATOR OF
      BEGIN
      1:
      YAZ:=YAZ CAT "=";
      2:
      YAZ:=YAZ CAT "(";
      3:
      YAZ:=YAZ CAT ")";
      4:
```

```
     YAZ:=YAZ CAT "<>";
     5:
     YAZ:=YAZ CAT "<=";
     6:
     YAZ:=YAZ CAT ">=";
     ELSE:BEGIN
          SAY 48"27E3";
          CURSOR (23,2,PBUF);
          SAY " Yanlis BAGINTI ; Lutfen duzeltin";
          SAY 48"27E603";
          WRYTE (REMOTE,PBUF);
          YAZ:=SAKLA;
          GO TO A1;
          END;
END CASE;
YAZ:=YAZ CAT " ";
IF TAKE(NSECONDOP ,1) = "*" THEN
   BEGIN
       ARA(NSECONDOP,DONUS,2);
       IF DONUS=0   THEN
       BEGIN
          SAY 48"27E3";
          CURSOR(23,2,PBUF);
          SAY " Yanlis IKINCI ISLENEN ;Lutfen duzeltin";
          SAY 48"27E603";
          WRYTE (REMOTE,PBUF);
          YAZ:=SAKLA;
          GO TO A1;
       END;
   END;
YAZ:=YAZ CAT NSECONDOP;
YAZ:=YAZ CAT " ";
IF NSOPERATOR NEQ 0 THEN
           BEGIN
         . CASE NSOPERATOR OF
                     BEGIN
                     1:
                     YAZ:=YAZ CAT "AND%";
                     2:
                     YAZ:=YAZ CAT "OR%"
                     ELSE:BEGIN
                          SAY 48"27E3";
                          CURSOR (23,2,PBUF);
                     SAY " Yanlis BAGLAYICI BAGINTI ;Lutfen duzeltin ";
                      SAY 48"27E603";
                      WRYTE (REMOTE,PBUF);
                      YAZ:=SAKLA;
                      GO TO A1;
                      END;
           END CASE;
           NYAZARTIK;
           YAZ:=" ";
           GO TO A
           END;
A2:
NYAZARTIK      ;
NSOPSONU: ;
END;
$PAGE
PROCEDURE NDELETECREATEUPDATEP;
```

```
BEGIN
STRING SAKLA;
INTEGER DONUS;
INTEGER OK1;
LABEL A0;
LABEL A1;
LABEL A11;
LABEL A,A01;
A0:
YAZ:= " ";
NFORMS(1);
A01:
READ (REMOTE,(A17),YAPI);
IF YAPI = "                   " THEN
   BEGIN
        GERIYEDON:=1;
        GO TO A;
   END;
CASE KOMUT OF
           BEGIN
           3:
           YAZ:= "DEL ";
           4:
           YAZ:= "CR ";
           5:
           YAZ:= "U ";
END CASE;
IF TAKE(YAPI,1) = "$" THEN
   BEGIN
       ARA(YAPI,DONUS,1);
       IF DONUS=0   THEN
       BEGIN
          SAY 48"27E3";
          CURSOR(23,2,PBUF);
          SAY " Yanlis YAPI ;Lutfen duzeltin";
          SAY 48"27E603";
          WRYTE (REMOTE,PBUF);
          GO TO A01;
       END;
   END;
YAZ:=YAZ CAT YAPI;
SAKLA:=YAZ;
IF KOMUT = 3   THEN
   BEGIN
       NYAZARTIK;
       GO TO A;
   END;
OK1:=0;
A11:
NFORMS(2);
A1:
READ(REMOTE,(A24,I1,A24,I1),NFIRSTOP,NFOPERATOR,NSECONDOP,NSOPERATOR);
IF NFIRSTOP = "                           " OR
   NSECONDOP = "                           " THEN
        BEGIN
            IF OK1 = 0 THEN
                GO TO AO
            ELSE
               BEGIN
               SAY 48"27E3";
```

```
              CURSOR (23,2,PBUF);
              SAY " Lutfen islenenleri giriniz";
              SAY 48"27E603";
              WRYTE (REMOTE,PBUF);
               GO TO A1;
              END
       END;
OK1:=1;
YAZ:=YAZ CAT " ";
IF TAKE(NFIRSTOP ,1) = "$" THEN
   BEGIN
       ARA(NFIRSTOP ,DONUS,2);
       IF DONUS=0   THEN
       BEGIN
          SAY 48"27E3";
          CURSOR(23,2,PBUF);
          SAY " Yanlis BIRINCI ISLENEN; Lutfen duzeltin";
          SAY 48"27E603";
          WRYTE (REMOTE,PBUF);
          YAZ:=SAKLA;
          GO TO A1;
       END;
   END;
YAZ:=YAZ CAT NFIRSTOP;
YAZ:=YAZ CAT " ";
IF NFOPERATOR = 1 THEN
   YAZ:= YAZ CAT "="
ELSE
    BEGIN
          SAY 48"27E3";
          CURSOR (23,2,PBUF);
          SAY " Yanlis BAGINTI ; Lutfen duzeltin";
          SAY 48"27E603";
          WRYTE (REMOTE,PBUF);
        . YAZ:=SAKLA;
          GO TO A1;
    END;
YAZ:=YAZ CAT " ";
IF TAKE(NSECONDOP ,1) = "$" THEN
   BEGIN
       ARA(NSECONDOP,DONUS,2);
       IF DONUS=0   THEN
       BEGIN
          SAY 48"27E3";
          CURSOR(23,2,PBUF);
          SAY " Yanlis IKINCI ISLENEN ;Lutfen duzeltin";
          SAY 48"27E603";
          WRYTE (REMOTE,PBUF);
          YAZ:=SAKLA;
          GO TO A1;
       END;
   END;
YAZ:=YAZ CAT NSECONDOP;
YAZ:=YAZ CAT " ";
IF NSOPERATOR NEQ 0 THEN
           BEGIN
           CASE NSOPERATOR OF
                     BEGIN
                     1:
                     YAZ:=YAZ CAT ",%";
```

```
                       ELSE:BEGIN
                            SAY 48"27E3";
                            CURSOR (23,2,PBUF);
                       SAY " Yanlis BAGLAYICI BAGINTI ;Lutfen duzeltin ";
                        SAY 48"27E603";
                        WRYTE (REMOTE,PBUF);
                        YAZ:=SAKLA;
                        GO TO A1;
                        END;
           END CASE;
           NYAZARTIK;
           YAZ:=" ";
           SAKLA := " ";
           GO TO A11;
           END;
NYAZARTIK;
A: ;
END;
$PAGE
PROCEDURE NQUITNEXTP;
BEGIN
LABEL A1,A01;
INTEGER DONUS;
CASE KOMUT OF
           BEGIN
           6:
           YAZ:= "N";
           7:
           YAZ:= "Q";
           18:
           YAZ:= "R";
           END;
IF KOMUT = 7 THEN  GO TO A1;
NFORMS(1);
A01:
READ (REMOTE,<A17>,YAPI);
YAZ:= YAZ CAT " ";
IF TAKE(YAPI,1) = "$" THEN
   BEGIN
       ARA(YAPI,DONUS,1);
       IF DONUS=0   THEN
       BEGIN
          SAY 48"27E3";
          CURSOR(23,2,PBUF);
          SAY " Yanlis YAPI ;Lutfen duzeltin";
          SAY 48"27E603";
          WRYTE (REMOTE,PBUF);
          GO TO A01;
       END;
   END;
       YAZ:= YAZ CAT YAPI;
A1:
NYAZARTIK;
END;
$PAGE
PROCEDURE NSHOWP;
BEGIN
LABEL A0, A1,A01,DATASETLI,DEFINELI,MAPLI;
LABEL SON,A2,A02,A3;
INTEGER DONUS,I,II,SECENEK;
```

```
STRING IA;                                              29003190
NFORMS(8);                                              29003195
AO:                                                     29003200
READ(REMOTE,(I1),SECENEK);                              29003205
CASE SECENEK OF                                         29003210
     BEGIN                                              29003215
        1:                                              29003220
        GO TO DATASETLI;                                29003225
        2:                                              29003230
        GO TO DEFINELI;                                 29003235
        3:                                              29003240
        GO TO MAPLI;                                    29003245
        ELSE:                                           29003250
            BEGIN                                       29003255
                 SAY 48"27E3";                          29003260
                 CURSOR (23,2,PBUF);                    29003265
                 SAY " Yanlis KOD ;Lutfen duzeltin ";   29003270
                 SAY 48"27E603";                        29003275
                 WRYTE (REMOTE,PBUF);                   29003280
                 GO TO AO;                              29003285
            END;                                        29003290
    END;                                                29003295
DATASETLI:                                              29003300
NFORMS (1);                                             29003305
AO1:                                                    29003310
READ (REMOTE,(A17),YAPI);                               29003315
IF YAPI = " " THEN                                      29003320
   BEGIN                                                29003325
   YAZ := "SH DA";                                      29003330
   GO A1;                                               29003335
   END;                                                 29003340
IF TAKE(YAPI,1) = "$" THEN                              29003345
   BEGIN                                                29003350
       ARA(YAPI,DONUS,1);                               29003355
       IF DONUS=0   THEN                                29003360
       BEGIN                                            29003365
          SAY 48"27E3";                                 29003370
          CURSOR(23,2,PBUF);                            29003375
          SAY " Yanlis YAPI ;Lutfen duzeltin";          29003380
          SAY 48"27E603";                               29003385
          WRYTE (REMOTE,PBUF);                          29003390
          GO TO AO1;                                    29003395
       END;                                             29003400
   END;                                                 29003405
   YAZ := "SH ALL OF " CAT YAPI;                        29003410
A1:                                                     29003415
NYAZARTIK;                                              29003420
GO TO SON;                                              29003425
DEFINELI:                                               29003430
NFORMS (7);                                             29003435
AO2:                                                    29003440
READ (REMOTE,(A17),YAPI);                               29003445
IF YAPI = " " THEN                                      29003450
   BEGIN                                                29003455
   YAZ := "SH DEF";                                     29003460
   GO A2;                                               29003465
   END;                                                 29003470
IF TAKE(YAPI,1) = "$" THEN                              29003475
   BEGIN                                                29003480
       ARA(YAPI,DONUS,1);                               29003485
       IF DONUS=0   THEN
       BEGIN
          SAY 48"27E3";
          CURSOR(23,2,PBUF);
          SAY " Yanlis TANIMLAMA ADI ;Lutfen duzeltin";
          SAY 48"27E603";
          WRYTE (REMOTE,PBUF);
          GO TO A02;
       END;
   END;
   YAZ := "SH DEF " CAT YAPI;
A2:
NYAZARTIK;
GO TO SON;
MAPLI:
II:=1;
A3:
SAY 48"OC";
FOR I:=2 STEP 1 WHILE I< 19 AND II<NUPOINTER+1  DO
BEGIN
     CURSOR (1,2,PBUF);
     SAY NU[II,1];
     CURSOR (1,35,PBUF);
     SAY "--)";
     CURSOR (1,40,PBUF);
     SAY NU[II,2];
     II:=II+1;
END;
CURSOR(1,1,PBUF);
WRYTE (REMOTE,PBUF);
IF II)NUPOINTER THEN GO TO SON;
READ(REMOTE,<A1>,IA);
IF IA="K" THEN GO TO SON;
GO TO A3;
SON:
END;
$PAGE
PROCEDURE NREPORTP;
BEGIN
STRING KBASLIK,
       KDEGISKEN,
       KONT,
       BASLIK,
       UZUNLUK,
       POZISYON,
       DEVAM;
INTEGER NTUR,
        NCONT;
INTEGER DONUS;
LABEL NRS;
LABEL A,A0,A01,A011,A02 ,A001;
A011:
NFORMS(1);
A01:
READ (REMOTE,<A17>,YAPI);
IF YAPI = "                     " THEN
   BEGIN
        GERIYEDON:=1;
        GO TO NRS;
   END;
```

```
IF TAKE(YAPI,1) = "$" THEN
   BEGIN
       ARA(YAPI,DONUS,1);
       IF DONUS=0   THEN
       BEGIN
          SAY 48"27E3";
          CURSOR(23,2,PBUF);
          SAY " Yanlis YAPI ;Lutfen duzeltin";
          SAY 48"27E603";
          WRYTE (REMOTE,PBUF);
          GO TO A01;
       END;
   END;
YAZ:="S VIA " CAT YAPI;
IF KOMUT=10 THEN
   YAZ:=YAZ CAT ";%";
NYAZARTIK;
NFORMS(3);
READ (REMOTE,(A40),BASLIK);
IF TAKE (BASLIK,1)=" " THEN GO TO A001;
YAZ:="TITLE PAGE " CAT 48"7F" CAT BASLIK CAT 48"7F";
IF KOMUT=10 THEN
   YAZ:=YAZ CAT ";%"       ;
NYAZARTIK;
A001:
NCONT:=0;
NTUR:=1;
A0:
NFORMS(4);
A:
READ (REMOTE,(A30,A17,A3,A2,A1,A1),KBASLIK,KDEGISKEN,UZUNLUK,
           POZISYON,KONT,DEVAM);
IF TAKE (KBASLIK,1) = " " AND
   TAKE (KDEGISKEN,1) = " " THEN
   GO TO A011;
IF KDEGISKEN="                   " THEN
   BEGIN
        SAY 48"27E3";
        CURSOR (23,2,PBUF);
        SAY " KOLON DEGISKENI gerekli ; Lutfen duzeltin ";
        SAY 48"27E603";
        WRYTE (REMOTE,PBUF);
        GO TO A;
   END;
IF TAKE(KDEGISKEN ,1) = "$" THEN
   BEGIN
       ARA(KDEGISKEN ,DONUS,1);
       IF DONUS=0   THEN
       BEGIN
          SAY 48"27E3";
          CURSOR(23,2,PBUF);
          SAY " Yanlis KOLON DEGIŞKENi;Lutfen duzeltin";
          SAY 48"27E603";
          WRYTE (REMOTE,PBUF);
          GO TO A;
       END;
   END;
IF NOT ( KONT="E" OR KONT =" ") THEN
               BEGIN
                   SAY 48"27E3";
```

```
                    CURSOR (23,2,PBUF);                                          29004080
           SAY " KONTROL OGESI (E/ ) olabilir ; Lutfen duzeltin ";               29004085
                    SAY 48"27E603";                                              29004090
                    WRYTE (REMOTE,PBUF);                                         29004095
                    GO TO A;                                                     29004100
                END;                                                             29004105
IF NOT (DEVAM="E" OR DEVAM=" ") THEN                                             29004110
                BEGIN                                                            29004115
                    SAY 48"27E3";                                                29004120
                    CURSOR (23,2,PBUF);                                          29004125
                    SAY " DEVAM KODU (E/ )olabilir; Lutfen duzeltin ";           29004130
                    SAY 48"27E603";                                              29004135
                    WRYTE (REMOTE,PBUF);                                         29004140
                    GO TO A;                                                     29004145
                END;                                                             29004150
IF KONT="E" AND NCONT=1 THEN                                                     29004155
                BEGIN                                                            29004160
                    SAY 48"27E3";                                                29004165
                    CURSOR (23,2,PBUF);                                          29004170
                    SAY " KONTROL OGESI girilemez ; Lutfen duzeltin ";           29004175
                    SAY 48"27E603";                                              29004180
                    WRYTE (REMOTE,PBUF);                                         29004185
                    GO TO A;                                                     29004190
                END;                                                             29004195
IF NTUR=1 THEN                                                                   29004200
   BEGIN                                                                         29004202
   YAZ := "REPO ";                                                               29004205
   NTUR:= 2;                                                                     29004206
   END                                                                           29004208
ELSE                                                                             29004209
   YAZ:= " ";                                                                    29004210
IF KONT NEQ "E" THEN                                                             29004213
   NCONT:= 1;                                                                    29004215
IF TAKE( KBASLIK,1)  NEQ " "  THEN                                               29004220
   YAZ := YAZ CAT KBASLIK CAT "=";                                               29004225
YAZ:=YAZ CAT " " CAT KDEGISKEN;                                                  29004230
IF TAKE( POZISYON,1)  NEQ " " THEN                                               29004255
   YAZ:=YAZ CAT " @ " CAT POZISYON;                                              29004260
IF TAKE( UZUNLUK,1)  NEQ " " THEN                                                29004265
   YAZ:=YAZ CAT " # " CAT UZUNLUK;                                               29004270
IF DEVAM = "E" THEN                                                              29004275
   YAZ := YAZ CAT " ,%"                                                          29004280
ELSE                                                                             29004285
   IF KOMUT=10 THEN                                                              29004290
      YAZ:=YAZ CAT ";%";                                                         29004295
NYAZARTIK;                                                                       29004300
IF DEVAM = "E" THEN                                                              29004305
   GO AO;                                                                        29004310
NFORMS(1);                                                                       29004315
A02:                                                                             29004320
READ (REMOTE,(A17),YAPI);                                                        29004325
IF TAKE(YAPI,1) = "$" THEN                                                       29004330
   BEGIN                                                                         29004335
       ARA(YAPI,DONUS,1);                                                        29004340
       IF DONUS=0   THEN                                                         29004345
       BEGIN                                                                     29004350
          SAY 48"27E3";                                                          29004355
          CURSOR(23,2,PBUF);                                                     29004360
          SAY " Yanlis YAPI ;Lutfen duzeltin";                                   29004365
          SAY 48"27E603";                                                        29004370
          WRYTE (REMOTE,PBUF);
          GO TO A02;
       END;
   END;
YAZ:="G REP VIA " CAT YAPI;
NYAZARTIK;
NRS:;
END;
$PAGE
PROCEDURE NTANIMLAP;
BEGIN
INTEGER SON;
LABEL NTS,OKU,CHECK;
INTEGER I;
STRING NBIR,NIKI;
NFORMS(5);
OKU:
READ (REMOTE,<A30,A30>,NBIR,NIKI);
IF TAKE(NBIR,1)= " " THEN GO TO NTS;
NUPOINTER:=NUPOINTER + 1;
IF NUPOINTER>100 THEN
                BEGIN
                    SAY 48"27E3";
                    CURSOR (23,2,PBUF);
                    SAY " 100 den fazla TANIMLAMA olamaz";
                    SAY 48"27E603";
                    WRYTE (REMOTE,PBUF);
                    WHEN (5);
                    GERIYEDON:=1;
                    GO TO NTS;
                END;
I:=1;
CHECK:
IF NU[I,1] = NBIR THEN
       BEGIN
                    SAY 48"27E3";
                    CURSOR (23,2,PBUF);
         SAY " Ayni isim daha once TANIMLAnmis;Lutfen duzeltin ";
                    SAY 48"27E603";
                    WRYTE (REMOTE,PBUF);
                    GO TO OKU;
      END;
I:=I+1;
IF I < NUPOINTER + 1 THEN
   GO CHECK;
NU[NUPOINTER,1]:=NBIR;
NU[NUPOINTER,2]:=NIKI;
NTS:;
END;
$PAGE
PROCEDURE NTANIMLAATP;
BEGIN
INTEGER SON;
LABEL NTS,OKU,CHECK;
INTEGER I;
STRING NBIR,NIKI;
NFORMS(5);
OKU:
READ (REMOTE,<A30,A30>,NBIR,NIKI);
IF TAKE (NBIR,1)=" " THEN GO TO NTS;
```

```
I:=1;
CHECK:
IF NU[I,1] = NBIR AND NU[I,2]=NIKI THEN
       BEGIN
                    NU[I,1]:="                    ";
                    NU[I,2]:="                    ";
                    GO TO NTS;
       END;
I:=I+1;
IF I < NUPOINTER + 1 THEN
   GO CHECK;
SAY 48"27E3";
CURSOR (23,2,PBUF);
SAY " Ayni isim daha once TANIMLAnmamis;Lutfen duzeltin ";
SAY 48"27E603";
WRYTE (REMOTE,PBUF);
GO TO OKU;
NTS:;
END;
$PAGE
PROCEDURE NMAPSAVERESTOREP;
BEGIN
STRING DOSYA;
IF KOMUT=16 THEN
   BEGIN
        WRITE(MAPSAVEW,<I3>,NUPOINTER);
        FOR I :=1 STEP 1 UNTIL NUPOINTER DO
            WRITE (MAPSAVEW,<A30,A30>,NU[I,1],NU[I,2]);
        LOCK (MAPSAVEW,CRUNCH);
   END
ELSE
   BEGIN
        READ (MAPSAVER,<I3>,NUPOINTER);
        FOR I :=1 STEP 1 UNTIL NUPOINTER DO
        .   READ  (MAPSAVER,<A30,A30>,NU[I,1],NU[I,2]);
        CLOSE(MAPSAVER);
   END;
END;
$PAGE
PROCEDURE NDEFINEP;
BEGIN
STRING DEFADI;
LABEL NRS;
NFORMS(7);
READ (REMOTE,<A17>,DEFADI);
IF DEFADI  = "                 " THEN
   BEGIN
        GERIYEDON:=1;
        GO TO NRS;
   END;
YAZ:= "DEF " CAT DEFADI CAT "=%";
NYAZARTIK;
NREPORTP;
NRS:;
END;
$PAGE
PROCEDURE NCLEARP;
BEGIN
STRING DEFADI;
LABEL NRS;
```

```
NFORMS(7);
READ(REMOTE,(A17),DEFADI);
IF DEFADI  = "                         " THEN
   BEGIN
        GERIYEDON:=1;
        GO TO NRS;
   END;
YAZ:="C DEF " CAT DEFADI;
NYAZARTIK;
NRS:;
END;
$PAGE
PROCEDURE NSAVERESTOREP;
BEGIN
STRING DOSYA;
LABEL A;
NFORMS(6);
A:
READ (REMOTE,(A17),DOSYA);
IF TAKE(DOSYA,1)=" " THEN
                BEGIN
                    SAY 48"27E3";
                    CURSOR (23,2,PBUF);
                    SAY " DOSYA ADI gerekli;Lutfen duzeltin ";
                    SAY 48"27E603";
                    WRYTE (REMOTE,PBUF);
                    GO TO A;
                END;
IF KOMUT=12 THEN
   BEGIN
        YAZ:= "SA ";
        YAZ:= YAZ CAT DOSYA;
        NYAZARTIK;
   END
ELSE
   BEGIN
        YAZ:= "RES ";
        YAZ:=YAZ CAT DOSYA;
        NYAZARTIK;
   END;
END;
PROCEDURE NPRINTERP;
BEGIN
LABEL A,OK;
INTEGER AC ,KAPA;
NFORMS(9);
A:
READ (REMOTE,(I1,I1),AC,KAPA);
IF ( AC=0 AND KAPA=0) OR (AC=1 AND KAPA=1) THEN
                BEGIN
                    SAY 48"27E3";
                    CURSOR (23,2,PBUF);
                    SAY " YANLIS KOD;Lutfen duzeltin ";
                    SAY 48"27E603";
                    WRYTE (REMOTE,PBUF);
                    GO TO A;
                END;
IF AC=1 THEN
   BEGIN
   YAZ:= "O P";
```

```
    GO OK;
    END;
 IF KAPA=1 THEN
    BEGIN
    YAZ:= "O T";
    GO OK;
    END;
SAY 48"27E3";
CURSOR (23,2,PBUF);
SAY " YANLIS KOD;Lutfen duzeltin ";
SAY 48"27E603";
WRYTE (REMOTE,PBUF);
GO TO A;
OK:
NYAZARTIK;
END;
$PAGE
PROCEDURE NMAIN;
BEGIN
LABEL A;
LABEL AL;
STRING DUMMYREAD;
 READ (REMOTE,(A1),DUMMYREAD);
AL:     NFORMS (0);
        GERIYEDON:=0;
A:
        READ (REMOTE,(I2),KOMUT)  ;
        CASE KOMUT OF
             BEGIN
             1:
             NSELECTDISPLAYP;
             2:
             NSELECTDISPLAYP;
             3:
             NDELETECREATEUPDATEP;
             4:
             NDELETECREATEUPDATEP;
             5:
             NDELETECREATEUPDATEP;
             6:
             NQUITNEXTP;
             7:
             NQUITNEXTP;
             8:
             NSHOWP;
             9:
             NREPORTP;
             10:
             NDEFINEP;
             11:
             NCLEARP;
             12:
             NSAVERESTOREP;
             13:
             NSAVERESTOREP;
             14:
             NTANIMLAP;
             NCONT2:=1;
             15:
             NTANIMLAATP;
```

```
            NCONT2:=1;
            16:
            NMAPSAVERESTOREP;
            NCONT2:=1;
            17:
            NMAPSAVERESTOREP;
            NCONT2:=1;
            18:
            NQUITNEXTP;
            19:
            NPRINTERP;
            20:
            NCONT:=0;
            ELSE :
                 BEGIN
                      SAY 48"27E3";
                      CURSOR (23,2,PBUF);
                      SAY " Yanlis KOD ; Lutfen duzeltin ";
                      SAY 48"27E603";
                      WRYTE (REMOTE,PBUF);
                      GO TO A;
                 END;
      END CASE;
      IF GERIYEDON=1 THEN GO TO AL;
END;
     REPLACE P:OB BY "HATA:";
           REPL "BELIRSIZ HATA"; %0
           REPL "YAZILIM HATASI"; %1
           REPL "OZEL KARAKTER BEKLENIYORDU"; %3
           REPL "AYRAC UYUMSUZLUGU     "; %4
           REPL "UYUSAN DIZI YOK    "; %5
           REPL "DEGISKEN COK BUYUK"; %6
           REPL "BU DEGISKEN HAKKINDA BILGI YOK      "; %7
           REPL "AT KOMUTU GEREKLI      "; %8
           REPL "TIRNAK EKSIK "; %9
           REPL "ISLENEN GEREKLI     "; %10
           REPL "YANLIS VEYA UYUMSUZ TANIMLAMA"; %11
           REPL "NITELIKLER YETERSIZ      "; %12
           REPL "GOSTER OPSIYONU YANLIS "; %13
           REPL "/=/ ISARETI GEREKLI"; %14
           REPL "/)/ ISARETI GEREKLI "; %15
           REPL "DEGISKEN VEYA LITERAL GEREKLI "; %16
           REPL "ILISKISEL ISLEC GEREKLI       "; %17
           REPL "BOOLEAN DEGIM GEREKLI        "; %18
           REPL "TAM SAYI GEREKLI"; %19
           REPL "TAM SAYI DAGILIM ARALIGI DISINDA "; %20
           REPL "VERI GRUBU GEREKLI "; %21
           REPL "HENUZ UYGULANMADI  "; %22
           REPL "DIZI GEREKLI  "; %23
           REPL "ARITMETIK DEGER GEREKLI "; %24
           REPL "FIIL GEREKLI "; %25
           REPL "BILINMEYEN DEGISKEN "; %26
           REPL "YANLIS SET OPSIYONU    "; %27
           REPL "FIIL ADI OLAMAZ     "; %28
           REPL "UYUMSUZ VERI GRUBU-SECME DEGIMI  "; %29
           REPL "DIZIN DAGILIM ARALIGI DISINDA "; %30
           REPL "DEFINES NESTED TOO DEEP";  % 31
           REPL "VERI TABANI OGE ADI OLAMAZ    "; %32
           REPL "DEFINE PARAMETER MIS-MATCH"; % 33
           REPL "DEKLERASYON LIMITI ASILDI ";  %34
REPL "CUMLE SONU BEKLENIYORDU  "; % 35                                30073400
REPL "HICBIR YAPI SECILMEMIS "; % 36                                  30073800
REPL "YAPI SECILMEMIS         "; % 37                                 30074200
REPL "GOMME YAPI OLMALIYDI       "; % 38                              30074600
REPL "SAHIP KAYDI YOK               "; % 59                           30075000
REPL "DEGISKEN TEK DEGIL   "; % 40                                    30075400
REPL "VERI OGESI GEREKLI "; % 41                                      30075800
REPL "BELIRSIZ OPSIYON "; % 42                                        30076200
REPL "COMMA OR THEN EXPECTED"; % 43                                   30076600
REPL "SECME KOSULU GEREKLI           "; % 44                          30077000
REPL "SET OPSIYONU-VERI GRUBU UYUMSUZ"; % 45                          30077400
REPL "DEFINE OGESI DEGIL"; % 46                                       30077800
REPL "RECALL NOT DONE";  % 47                                         30078200
REPL "BAD $ SYNTAX"; % 48                                             30078600
REPL "IDENTIFIER NOT TEMPORARY SET ID"; %49                           30079000
REPL "DIZIN GEREKSIZ          "; % 50                                 30079400
REPL "DIZINLER FAZLA       "; % 51                                    30079800
REPL "DIZINLER YETERSIZ     "; % 52                                   30080200
REPL "OGE BU VERI GRUBUNDA DEGIL               "; % 53                30080600
REPL "VE,VEYA ,+,VEYA - GEREKLI "; % 54                               30081000
REPL "ISLECLER KARISTIRILAMAZ"; % 55                                  30081400
REPL "SADECE IKI ISLEC GEREKLI  "; % 56                               30081800
REPL "VERI GRUPLARI KARISTIRILAMAZ "; % 57                            30082200
REPL "DIZIN BASIT OGE OLMALI         "; % 58                          30082600
REPL "TEMPORARY SET NOT INITIALIZED"; % 59                            30083000
REPL "EKRAN CIKTI ICIN YETERSIZ  ";  % 60                             30083400
REPL "DOSYA ADI YANLIS OLUSTURULMUS "; %61                            30083800
REPL "DIZIN SETI GEREKLI"; % 62                                       30084200
REPL "DONGU VAR     "; % 63                                           30084600
REPL "YAPI ADI GEREKLI     "; % 64                                    30085000
REPL "SECME DEGIMI UYUMSUZ              "; % 65                       30085400
REPL "MUST COMBINE DIFFERENT TEMP-SETS"; % 66                         30085800
REPL "GENERATE,VIRTUAL OR DEFINE EXPECTED";%67                        30086200
REPL "VIRTUAL NOT INITIALIZED";  %68                                  30086600
REPL "VTREE OVERFLOW";  % 69                                          30087000
REPL "TTD OVERFLOW";  % 70                                            30087400
REPL "SOL PARANTEZ GEREKLI";  %71                                     30087800
REPL "TUM OGELER AYNI AILEDEN OLMALI       ";  %72                    30088200
REPL "ARITMETIK DEGER GEREKLI ";  %73                                 30088600
REPL "SAHIP YAPI HENUZ SECILMEMIS     ";  %74                         30089000
REPL "MUST BE DESCENDENT OF PREVIOUS STRUCTURE";%75                   30089400
REPL "CANNOT EVALUATE FUNCTION (CHANGE STR)";  %76                    30089800
REPL "CANNOT SEARCH VIA THIS EMBEDDED DATASET"; %77                   30090200
REPL "VIRTUAL NO LONGER DEFINED";  % 78                               30090600
REPL "BILINMEYEN UC OZELLIGI    ";  % 79                              30091000
REPL "FAZLA BILGI                         "; % 80                     30091400
REPL "GLOBAL VERI OGELERINDE FONKSIYON OLAMAZ   ";                    30091800
REPL "GLOBAL VERI YOK, SECILEMEZ    "; %82                            30092200
REPL "DEFINE PARAMETERS CANNOT EXCEED 9";%83                          30092600
REPL "FONKSIYON VERI GRUBU ICERMELI ";%84                             30093000
REPL "BOOLEAN ITEM ILLEGAL IN RELATION"; %85                          30093400
REPL "FONKSIYON SONUCU OZYINELI     "; %86                            30093800
REPL "DIZIN GEREKLI      "; % 87                                      30094200
REPL "GUNLEME YAPILAMAZ                 "; % 88                       30094600
REPL "BU OGEYE DEGER YUKLENEMEZ                  "; %89               30095000
REPL "ESIT ISLECI GEREKLI            "; %90                           30095400
REPL "KAYIT KILITLENEMEDI  ";%91                                      30095800
REPL "ANAHTAR OGESI DEGERI YANLIS ";   %92                            30096200
REPL "REQUIRED ITEM IS NULL OR VERIFY CONDITION FAILED"; %93          30096600
REPL "VERI TABANI GERI DONDU ";     %94                               30097000
REPL "VARIABLE FORMAT ITEM NOT IN RECORD"; %95                    30097400
REPL "TANIMSIZ ARITMETIK DEGIM";%96                                30097800
REPL "DOGRU VEYA YANLIS GEREKLI "; %97                             30098200
REPL "GUNLEME LISTESI BOS "; %98                                   30098600
REPL "IKI DEGISIK VERI GRUBU BELIRLENMIS "; %99                    30099000
REPL "SORGU BU GOMME VERI GRUBUNU GUNLEYEMEZ";%100                 30099400
REPL "DATA OPSIYONLU SECME DE GUNLEME YAPILAMAZ ";                 30099800
REPL "SADECE OKUMA OGELERI SET EDILEMEZ ";  % 102                  30100200
REPL "DOSYA ISMI OLAMAZ          ";  % 103                         30100600
REPL "GIRILEN VERILER COK FAZLA ";   %104                          30101000
REPL "DATA OPTION ALLOWED IN CIRCULAR REF. ONLY";                  30101400
REPL "DIZIN DEGERI TANIMSIZ       "; % 106                         30101800
REPL "GENERATE NOT ALLOWED FOR ORDERED DATASET";%07                30102200
REPL "SIRALAMA ANAHTARI SAYISI EN COK 25 OLABILIR ";  % 108        30102600
REPL "BELIRTILEN OGE SIRALAMA ANAHTARI OLAMAZ"; % 109              30103000
REPL "ILLEGAL SORT SYNTAX"; %110                                   30103400
REPL "BAD SORT OPTION"; % 111                                      30103800
REPL "SORT NOT ALLOWED FOR ORDERED DATASET";%112                   30104200
REPL "FAZLA    PARAMETRE "; % 113                                  30104600
REPL "YETERSIZ SAYIDA PARAMETRE "; % 114                           30105000
REPL "INVALID FUNCTION ARGUMENT"; % 115                            30105400
REPL "YANLIS DIZIN DAGILIM ARALIGI "; % 116                        30105800
REPL "CANNOT REPEAT OWN STRUCTURE"; %117                           30106200
REPL "MAY NOT OCCUR IN DEFINE TEXT";                               30106600
REPL "MULTI-STMT DEFINE MUST CONTAIN COMPLETE STMTS";              30107000
REPL "PAGE EJECT ONLY ON CONTROL ITEMS";                           30110100
REPL "RAPOR COK BUYUK                    ";                        30110300
REPL "FAZLA /:/  ";                                                30110500
REPL "GRUP ADI OLAMAZ"; % 123                                      30110700
REPL "RAPOR LISTESINDE YOK ";                                      30110900
REPL "SAYISAL OGE OLMALI  ";                                       30111100
REPL "ALFABETIK OGE OLMALI ";                                      30111300
REPL "REPORT STATEMENT REQUIRED";                                  30111500
REPL "REPORT ITEMS NOT SPECIFIED";                                 30111700
REPL "COLUMN OVERLAP";                                             30111900
REPL "TOO MANY ITEMS IN REPORT LIST";                              30112100
REPL "NEW TEXT TOO BIG; CHANGE IGNORED";                           30112300
REPL "SILME YAPILAMAZ                    ";                        30112500
REPL "DELETE ILLEGAL FOR SELECT WITH DATA OPTION";                 30112700
REPL "YARAT VERI GRUBU ICIN OLMALI ";                              30112900
REPL "VERI GRUBU ICIN KAYIT TIPI BELIRLENMELI ";                   30113100
REPL "YANLIS KAYIT TIPI  ";                                        30113300
REPL "YARATMA YAPILAMAZ                 ";                         30113500
REPL "GOMME YAPILAR VEYA ILISKILI YAPILAR SILINEMEZ",              30113700
     "                 ";                                          30113800
REPL "SAYACI SIFIR OLMAYAN KAYIT SILINEMEZ                    ";   30114000
REPL "BOS KONTROLU SADECE /=/ VEYA /()/ ILE YAPILIR";              30114200
REPL "BOS KONTROLU SADECE VERI OGELERI ICIN GECERLI ";             30114400
REPL "OGE BOS OLAMAZ                                          ";   30114600
REPL "OGE BOS YAPILAMAZ            ";                              30114800
REPL "ERROR IN MASK HANDLING";                                     30115000
REPL "23 HANEDEN BUYUK ARITMETIK DEGER OLAMAZ             ";       30115200
REPL "EXTRACT STATEMENT REQUIRED";                                 30115400
REPL "EXTRACT ITEMS NOT SPECIFIED";                                30115600
REPL "YANLIS ALAN DEGERI ";                                        30115800
REPL "YANLIS ALAN BOYU DEGERI ";                                   30116000
REPL "YANLIS KAYIT BOYU DEGERI";                                   30116200
REPL "YANLIS BLOKBOYU DEGERI ";                                    30116400
REPL "BILINMEYEN DOSYA OZELLIGI ";                                 30116600
REPL "DUYARLIK COK BUYUK ";                                        30116800
```

```
          REPL "SCALE FACTOR TOO LARGE";                                        30117000
          REPL "/1/ GEREKLI             ";                                      30117200
          REPL "MUST BE BYTE BOUNDARY";                                         30117400
          REPL "TOO MANY ITEMS IN EXTRACT LIST";                                30117600
          REPL "YANLIS  FORMAT TIPI";                                           30118000
          REPL "YANLIS IC ISIM          ";                                      30118200
          REPL "INVALID ERROR ACTION";                                          30118400
          REPL "INVALID ERROR ACTION VALUE";                                    30118600
          REPL "BASLIK COK BUYUK                ";                              30118800
          REPL "BASLIK TA SAYFA NUMARASI YERI YOK ";                            30119000
          REPL "SECME DEGIMINDE EN FAZLA 47 KRITER OLABILIR";                   30119200
LABEL NREAD;                                                                    30177020
NREAD:                                                                          30177030
IF KAYITSAYISI > 0 THEN                                                         30177040
   BEGIN                                                                        30177050
       GB1:=READ(DISKTEN,FMT,IB);                                               30177060
       KAYITSAYISI:=KAYITSAYISI - 1;                                            30177070
       IF KAYITSAYISI=0 THEN CLOSE (DISKTEN,PURGE);                             30177080
   END                                                                          30177090
ELSE                                                                            30177100
   IF NCONT = 1 THEN                                                            30177110
      BEGIN                                                                     30177120
          NMAIN;                                                                30177130
          IF NCONT=0 THEN GO NREAD;                                             30177135
          IF NCONT2=0 THEN                                                      30177136
             BEGIN                                                              30177138
                LOCK (DISKE,CRUNCH);                                            30177140
                GB1:=READ(DISKTEN,FMT,IB);                                      30177150
                KAYITSAYISI:=KAYITSAYISI - 1;                                   30177160
                IF KAYITSAYISI=0 THEN CLOSE (DISKTEN,PURGE);                    30177170
                NCONT2:=0;                                                      30177174
             END                                                                30177176
          ELSE                                                                  30177178
             BEGIN                                                              30177179
                NCONT2:=0;                                                      30177180
                GO TO NREAD;                                                    30177181
             END;                                                               30177182
       END                                                                      30177185
   ELSE                                                                         30177190
         REPLACE OB BY "#GIRIS HATASI";                                         30182000
         MSGOUT(13);                                                            30183000
IF NCONT=0 THEN                                                                 30191010
   BEGIN                                                                        30191020
      IF IB="SORGU" THEN                                                        30191040
         BEGIN                                                                  30191050
            NCONT:=1;                                                           30191060
            REPLACE PBUF:BUF BY 48 "OC";                                        30191065
            GO TO NREAD;                                                        30191070
         END;                                                                   30191080
   END;                                                                         30191090
      REPLACE OB BY "#GIRIS HATASI";                                            30195760
      MSGOUT(13);                                                               30195770
%    REPLACE OB BY "#%";                                                        30199000
%    MSGOUT(2);                                                                 30200000
     REPLACE P:OB BY "*HANGI VERI GRUBU?";                                      31254000
        REPLACE OB BY "*BILGI ATLANDI. TANIMLAMA GEREKLI";                      31266800
        MSGOUT(33);                                                             31267000
          REPLACE OB BY "#GIRIS HATASI";                                        31268400
          MSGOUT(13);                                                           31268500
               REPLACE P:OB BY "*NUMARA YANLIS  DUZELTIN VEYA # G",             31274000
```

```
           "IRERER ISLEMI KESIN  ";                                             31274100
      REPLACE P:OB BY "*NUMARA GEREKLI DUZELTIN VEYA ",                         31280400
                "# GIRERER ISLEMI KESIN";                                       31260500
 " SORGU   " FOR 9,                                                             31464000
 REPLACE OB BY "#SORGU BASLIYOR";                                               31470000
 MSGOUT(15);                                                                    31471000
       REPLACE OB BY "VERI TABANI BOZULABILIR;GUNLEME YAPILIRKEN "              31477710
           "SISTEM HATASI OLUSTU ";                                             31477720
       MSGOUT(64);                                                              31477730
           WRITE(REMOTE,("CALISAMAZSINIZ"));                                    31508000
 REPLACE OB BY "#",  DBNAME FOR DBNAMESZ," HAZIR";                              31514000
 REPLACE QQ:ERRA BY "SAHIP    ";                                                32135000
 REPLACE QQ:QQ BY ") SECILMEMIS  ", 48"01";                                     32140000
REPLACE QQ:ERRA BY " ";                                                         32141600
REPLACE QQ:QQ BY " KAYDI YARATILDI ", 48"01";                                   32141660
 REPLACE QQ:ERRA BY " ";                                                        32147000
 REPLACE QQ:QQ BY " KAYDI SECILMEMIS ", 48"01";                                 32150000
 REPLACE QQ:ERRA BY "DEGER DAGILIM ARALIGI DISINDA ";                           32326000
 REPLACE OB BY "  OGELER:";                                                     40063000
 MSGOUT(9);                                                                     40064000
 REPLACE OB BY "  GRUPLAR:";                                                    40145000
 MSGOUT(10);                                                                    40146000
 REPLACE OB BY "     VERI GRUPLARI:";                                           40173000
 REPLACE TTX[T] BY " VERI GRUBU";                                               40332000
 T := *+11;                                                                     40333000
      REPLACE OB BY "  GLOBAL OGELER:";                                         40442000
      MSGOUT(16);                                                               40443000
    REPLACE OB BY "*GLOBAL YOK";                                                40455000
    MSGOUT(11);                                                                 40456000
      REPLACE OB BY "*BOS";                                                     42060000
      MSGOUT(4);                                                                42061000
           REPLACE OB BY "# YOK  ";                                             44079000
   REPLACE OB BY "# EXTRACT HATALI BITIRILDI";                                  53542220
    REPLACE P:OB BY "** RAPOR ICIN KAYIT YOK    **";                            53693720
    REPLACE OB BY "# EXTRACT BASLIYOR";                                         53693840
      REPLACE P:P BY "-BOS GIRIS BEKLIYOR    ";                                 53742000
      REPLACE P:OB BY " RAPOR YAZILIYOR";                                       53756000
         REPLACE OB BY "# RAPOR KESILDI  ";                                     53776200
         REPLACE OB BY "# EXTRACT KESILDI";                                     53776300
         MSGOUT(17);                                                            53776320
    REPLACE OB BY "# EXTRACT BITTI";                                            53803500
    MSGOUT(15);                                                                 53803600
 REPLACE P:OB BY "# RAPOR BITTI";                                               53811000
           REPLACE OB BY "#DURDU  ";                                            58006500
           REPLACE OB BY "#DURDU.  ";                                           58027000
           REPLACE OB BY "#DURDU  ";                                            58050000
           REPLACE OB BY "IYI KI YOK";                                          63428000
           REPLACE OB BY "#BEKLE";                                              64903000
           MSGOUT(6);                                                           64904000
                REPLACE OB BY "#YOK";                                           65065000
                MSGOUT(4);                                                      65066000
           REPLACE OB BY "#DURDU  ";                                            65441400
      REPLACE OB BY"#BEKLE"; MSGOUT(6);                                         67285000
      REPLACE OB BY "#HELP DOSYASI YOK";                                        71541000
      MSGOUT(17);                                                               71542000
      REPLACE OB BY "#YOK";                                                     72102000
      MSGOUT(4);                                                                72103000
      REPLACE OB BY "#DAHA YOK";                                                73162000
      MSGOUT(9);                                                                73163000
 REPLACE QQ:OB BY " ";                                                          76068400
 REPLACE QQ:QQ BY " SECILMIS KAYDI GUNLENDI" ;                          76068600
REPLACE QQ:OB BY " ";                                                   76562800
REPLACE QQ:QQ BY " SECILMIS KAYDI ATILDI ";                             76563200
REPLACE QQ:OB BY " ";                                                   76935000
REPLACE QQ:QQ BY " KAYIDI YARATILDI ";                                  76935400
        "***DIKKAT-CHECK CONTROL AND/OR REPORT SUMMARY";                77248000
 REPLACE OB BY "#TANIMLAMALAR DEFAULT DOSYA DA SAKLANMADI";             79005000
 MSGOUT(41);                                                            79006000
 REPLACE OB BY "#SORGU BITIYOR";                                        80053400
```

APPENDIX E LISTING OF MODIFICATIONS TO BUILDSORGU

```
1 $MERGE
16013000    REPLACE P:SP BY "****** HATA ****** ";
16016000        REPLACE P:P BY " TABLO TASMASI ";
16017000        REPLACE P:P BY " YANLIS VERI TABANI ";
16018000        REPLACE P:P BY " YANLIS VERI TABANI ADI  ";
16019000        REPLACE P:P BY " VERI GRUPLARI 255 I ASTI       ";
16020000        REPLACE P:P BY "YANLIS OPSIYON";   % 4
16021000        REPLACE P:P BY "YAPI IKINCI DEFA GECIYOR ";   % 5
16022000        REPLACE P:P BY "PROPERTIES VE DESCRIPTION SEVIYELERI UYUMSUZ\
         \ ";
16023000        REPLACE P:P BY " BUILDING YENIDEN YARATIN LUTFEN ";  % 7
16025000            REPLACE SP BY "******* INPUT HATALI   ****** ",48"00"; \
         \%8
16029000        REPLACE P:P BY "YAZILIM HATASI";  % 9
16030000        REPLACE P:P BY "/:/ GEREKLI    ";  % 10
16031000        REPLACE P:P BY "YANLIS KART GIRISI ";  % 11
16032500        REPLACE P:P BY "DIRECTORY BOYU COK BUYUK "; % 13
16032600        REPLACE P:P BY "RESTART VERI GRUBU YOK         "; % 14
16032700        REPLACE P:P BY "VERI TABANI YAPI ISIMLERI SECILMIS OLMAMALI \
         \";
16050500    NOZIP:=TRUE ;
16051000    REPLACE R:DBNAME BY "HANGI VERI TABANI ?";
20138600      REPLACE Q:A BY "INTERPRETER ADI "
20139000      REPLACE Q:A BY "SORGU PROGRAM ISMI  ";
20141100    ELSE REPLACE Q:Q BY " (DEFAULT ICIN BOS)?";
20150000            REPLACE Q:A BY " DEFAULT ISIM KULLANILAMAZ ";
20152000            REPLACE Q:A BY "ISMI YENIDEN GIRIN VEYA BITIRMEK ICIN *\
         \";
20166510    REPLACE Q:A BY "HANGI QUEUE (DEFAULT ICIN BOS)?";
20166660        REPLACE Q:A BY "YANLIS QUEUE DEGERI ";
20356350    REPLACE Q:A BY " ";
20356480      REPLACE Q:Q BY "SADECE SORGULAMA ";
20356550      REPLACE Q:Q BY "GUNLEME";
20356650      REPLACE Q:Q BY "YARATMA";
20356750      REPLACE Q:Q BY "ATMA  ";
20357000    REPLACE Q:Q BY " (EVET VEYA HAYIR)?";
20362000    IF DELTA(Q,R)=4 AND Q="EVET" THEN
20362200    ELSE IF NOT(DELTA(Q,R)=5 AND Q="HAYIR") THEN
21006000    REPLACE P:A BY "DEFINITION DOSYA ADI (DEFAULT BOS)?";
32023000    LABEL ENDL,XITP,RETRY,HATALI ;
32025000    REPLACE Q:A BY "ASAGIDAKI VERI GRUBU ICIN    :";
32027000    REPLACE Q:A BY " D SORGU DISINDA KALSIN";
32029000    REPLACE Q:A BY " K SORGU ICINE KABUL";
32031000    REPLACE Q:A BY " B SECIMLER BITTI  ";
32033000    REPLACE Q:A BY " * PROGRAM BITIRMEK ICIN ";
32045500  HATALI:
32049000            IF Q = "D" THEN GO ENDL;
32050000            IF Q = "B" THEN GO XITP;
32052000            IF Q NEQ "K" THEN
32055000                "??? YENIDEN GIRIN VEYA * LA PROGRAMI BITIRIN ";
32057000                GO HATALI;
33031000    REPLACE Q:A BY "MANTIKSAL VERI TABANI ADI ?";
70010314     REPLACE Q:A BY "HANGI YAZILIM ?";
70010318     REPLACE Q:A BY "1 SORGU    ";
70030000 REPLACE Q:A BY "HANGI OPSIYON ";
70032000 REPLACE Q:A  BY " 1 TUM VERI TABANI" ;
70034000 REPLACE Q:A BY " 2 SECILMIS VERI GRUPLARI ";
70038000    REPLACE Q:A BY " 3 MANTIKSAL VERI TABANI ";
90021000       REPLACE P:P BY "OBJECT/SORGU";
90049600      REPLACE P:P BY "OBJECT/SORGU";
```

BIBLIOGRAPY

1. Wiederhold, Gio, Database Design, McGraw-Hill Kogakusha, Ltd., 1977.

2. Murdick, Robert G., Ross, Joel E., Information Systems For Modern Management, Prentice-Hall, Inc., 1975.

3. Date, C.J., An Introduction to Database Systems, 2E, Addison-Wesley Publishing Company, 1979.

4. _____, Datapro 70 the EDP Buyer's Bible, Datapro Research Corporation, 1979.

5. _____, Introduction to B5000/B6000/B7000 DMS II INQUIRY, Burroughs Corporation, 1981.

6. Lancaster, F. Wilfrid, Information Retrieval Systems: Characteristics, Testing and Evaluation, John Wiley Sons, Inc., 1979.

REFERENCES NOT CITED

1. Jensen, Randall W., Tonies, Charles C., Software Engineering, Prentice Hall, Inc., 1979.

2. DMS II DASDL, Burroughs Corporation, 1981.