HUMAN RESOURCE SCHEDULING : AN APPLICATION OF LINEAR PROGRAMMING IN AN AUDIT FIRM

by

I.Rusen Kopmaz

B.S. in Istanbul Teknik University , 1986

Submitted to the Institute for Graduate Studies in

Social Sciences in partial fulfillment of the

requirements for the degree of Master's of Arts

in

Business Administration

Bogazici University

1990

APPROVED BY

Prof.Dr.Ceyhan Uyar

Prof.Dr.Seyhan Tugcu

Prof.Dr.Muzaffer Bodur

DATE OF APPROVAL,SEPTEMBER 1990

ACKNOWLEDGEMENTS

This study was conducted under the supervision of Prof.Dr.Ceyhan Uyar to whom I would like to express my deep gratitude for her valuable contributions and guidance.I am especially indebted to Prof.Dr.Seyhan Tugcu for the encouragement and special assistance which she provided on so many occasions.

TABLE OF CONTENTS

Page

Page

ABSTRACT

In a developing business society,human resources become the most valuable asset of the firms day by day.Considering this fact,most of the companies do not hesitate to make huge investments in this area. On the other hand, effective and efficient utilization of these rare and expensive resources creates problems for managements.

In this study,I aim to solve human resource scheduling problems that a worldwide auditing firm face by structuring a linear programming model. In general,auditing firms perform audit work on companies' financial statements as at year-end or any time in order to prepare a report which indicates whether the company's records were done in accordance with Generally Accepted Accounting Principles and legal requirements.At each step of the audit work, an intensive involvement of human resources from different levels is necessary.Due to the importance of human resources' contribution to the work done,auditing firms invest big amounts for human resources and related systems.Effective and efficient utilization of human resources which is considered an important management problem

for service firms,can be achieved by planning the allocation of resources to works for specific periods through out the whole working period.In an auditing firm,efficiency of any resource is measured by chargebility percentage which can be calculated for a period by dividing the total working hours of the resource that can be invoiced to the client by the total hours that the auditing firm paid for the resource.Auditing firm management aims to maximize chargebility for each one of its resources. In this study, a 0-1 Linear Integer Programming (IP) model which aims to maximize chargibility was formulated.The integer programming problem was solved by an implicit enumeration algorithm after certain modifications. A computer program in Fortran 77 was written for the algorithm and executed on a CDC computer system. At the end of the process, jobs that maximize the chargibility when they are scheduled were determined.

OZET

Gun gectikce rekabete yonelen is dunyasinda, ozellikle hizmet sektorunde insan kaynaklarinin onemi her gecen gun biraz daha artmaktadir.Bu onemi anlayan sirketler insan kaynaklarina buyuk yatirimlar yapmaktadirlar. Diger yandan bu az bulunur ve pahali kaynaklarin verimli ve etkin kullanilmasi sirket yonetimlerinin karsilastigi onemli problemlerden biridir.

Bu calismada dunya capinda hizmet veren bir bagimsiz denetim firmasinin insan kaynaklari planlanmasinda karsilastigi problemin , bir lineer tamsayi programlama modeli olusturularak cozulmesi amaclanmistir. Bagimsiz denetim firmalari,sirketlerin genelde sene sonu veya herhangi bir tarih itibariyle mali kayitlarini denetler ve bu kayitlarin gerek uluslararasi kabul gormus muhasebe ilkelerine gerekse de yasal yukumluluklere uygunlugu konusunda fikir bildiren bir rapor hazirlar. Butun bu kayitlarin incelenmesi, cesitli analizlerin ve testlerin yapilmasi, raporun hazirlanmasi cesitli seviyedeki insan kaynaklarinin katilimlariyla basarilabilir.Yapilan is bireylerin ve sistemin gelismis

olmasini gerektirdiginden, sirketler bu konudaki yatirimlardan kacinmamaktadirlar.Eldeki kaynaklarin kit ve pahali olmasindan dolayi bu kaynaklarin en verimli ve etkin sekilde kullanimi onem kazanmistir.Etkin ve verimli kullanim ise, butun calisma donemi boyunca hangi kaynaklarin hangi islerde ve hangi zamanlarda kullanilacaginin planlanmasiyla basarilabilir.Bagimsiz denetim firmalarinda kaynaklarin verimliligi, herhangi bir donemde kaynagin firmaya para kazandiran calisma saatlerinin, kaynagin firmadan ucretini aldigi saatlere bolumuyle bulunur.Bagimsiz denetim firma yonetimlerinin amaci, her bir calisanlari icin verimliligi maksimize etmektir.

Bu calismada amac fonksiyonu verimliligi maksimize eden bir 0-1 tamsayi programlama modeli olusturulmustur. Donald R.Plane ve Claude McMillan,Jr. tarafindan gelistirilen Ortuk Birerleme (Implicit Enumeration) algoritmasi cesitli duzenlemelerle modele uygulanmistir.Algoritma icin Fortran 77 dilinde yazilan bilgisayar programi CDC bilgisayar sisteminde calistirilmis ve sonucta calisma donemine yerlestirildiginde maksimum verimliligi saglayan islerin secimi yapilmistir.

INFORMATION ABOUT THE BUSINESS OF THE FIRM

Definition of an Audit:

An audit is an examination of financial statements or other data derived from a client's accounting processes for the purpose of expressing an opinion on certain characteristics of that data such as fair presentation and accuracy. Most commonly, auditors examine:

-A client's balance sheet as of one or more dates.

-The related statements of income and cash flows for the period(s) included in the balance sheets.

Purpose of an audit:

Testing financial statement assertions:

Users of an entity's financial statements--investors, creditors,lenders, regulatory agencies, ect.--normally want assurance from an independent party that the company's financial representations are credible.

The primary purpose of an audit is to provide this assurance by testing the six financial statement assertions that are made by management as follows:

Existence or occurance:

The assets and liabilities exist, and the transactions have occured as of or for the period ended on the dates specified.In other words, the company owns the assets and owns the liabilities as of the dates specified.

Completeness:

The financial statements are complete.All of the transactions and accounts that should be presented are included in the financial statements.

Accuracy:

Transactions and account balances are accurately recorded.

Valuation:

Assets are stated at realizable value, and liabilities are stated at expected settlement amounts.

Rights and obligations:

Assets are the rights, and liabilities are the obligations of the entity at the given date.

Presentation and disclosure:

Particular components of the financial statements have been properly classified, described, and disclosed.

History of Auditing:

Audit comes from the Latin word "audire" which means "to hear". Literally, auditing has existed for a long time.Before 1500, Byzantine rulers used two scripes to keep independent records of transactions.This made it easy to detect defalcations or embezzlements by comparing the two sets of records. Accuracy was not really an issue; the ruler was primarily concerned with fraud, a concern which was the focus of auditing until the twentieth century. During the early Middle Ages, merchants of Italien city states needed assurance about the transactions performed by the captains of their ships.The merchants hired auditors to ensure that money or goods were properly accounted for.

The Industrial Revolution began in the 1800s,and companies grew in size. Owners hired other people to manage the companies.These owners became concerned about the protection and growth of their investments.In Britain, in particular,groups of stockholders visited their companies to verify the accounting records.As a result of these visits,it became apparent that auditing required specialized knowledge and training.

Following the U.S. Civil War,the American economy expanded rapidly. As a result,Europeans invested heavily in U.S. companies.There was no accounting profession in the U.S at that time.(Although there were accountants, there was no professional organization or set of ethical standards to govern accounting practices.) European investors, mostly British,sent their own accountants to the U.S. to examine companies in which they had invested.This provided a starting point for many U.S. accounting firms, as Americans became exposed to this form of business and as some British firms opened permanent U.S. operations.

Around 1900, U.S. commercial banks began to use public accountants to audit companies obtaining bank financing.In 1903, United States Steel Corporation became the first American company to publish financial statements accompanied by an auditors' report.Other large companies followed this example.

The role of the auditor in the U.S. became even more important after the Great Depression and the passage of the Securities Exchange Laws of 1933 and 1934.These laws required audits of companies which were offering or had outstanding public securities and debt.

A similar evolution has occured throughout the industrialized world as third-party reporting has become more important in the capital formation process.Nowadays, in many countries laws require audits of all companies,both public and private,and also regulate the degree of financial statement disclosure and the form of the audit report.

In recent years,there has been an increasing interest in the detection of fraud and in internal controls,including internal controls for electronic data processing (EDP).More emphasis has also been placed on the auditor's social and public responsibility to the users of the financial statements.

The large international accounting firms not only perform audits,but they also provide many other services.Accounting firms assist clients in tax matters;provide management information consulting;perform studies and evaluations of internal controls,including EDP-related internal controls; and provide services related to financial planning,such as developing financial projections.These are just a few examples of the many types of services provided by accounting firms today.

Audit Team:

Each audit is planned and performed by a team under the overall supervision of the engagement partner (the audit partner).The size, composition and experience levels of the team are consistent with the nature, size and complexity of the assignment.Each member of the team is assigned tasks that he or she has adequate training and experience to complete.In addition to the engagement partner, an audit team typically consists of an audit manager, an in-charge person, and one or more staff accountants. An additional audit manager or senior who is a member of the Computer Audit Speciality Team (CASTS) is assigned to the audit team for clients that have complex computer systems.

Engagement Partner:

The engagement partner has overall responsibility for the quality of an audit and for the resulting Auditor's Report.In fulfilling these responsibilities the partner participates actively in the determination of the scope of the audit.

Engagement Manager:

The engagement manager plays a key role in assisting the engagement partner in discharging the firm's responsibility to a client and to the public. One of his or her major responsibilities is to participate in and supervise the progress of the audit.This is to ensure that potential audit problems are adressed and are brought to the attention of the engagement partner on a prompt basis.

In charge:

Usually a senior or a semi-senior is designated as being "in charge" of the day-to-day operations of the audit engagement.The in-charge normally participates in the development of the overall audit plan.He or she also supervises and assists the staff accountant in fulfilling their daily work responsibilities.

Staff Accountant:

A staff accountant, works with--and under the direction of the in-charge. Experienced assistants and assistants comprise the two level of staff accountant classification.In general, the staff accountants perform the following jobs:

--They meet with the client personel to determine the current status of the client's procedures

--They perform test of controls on a number of the client's accounting functions.

--They perform substantive tests of certain transactions and account balances.

--They prepare working-papers and documentation.

HUMAN RESOURCE SCHEDULING PROBLEM OF THE FIRM

Definition of the Problem:

The firm renders audit services over 100 clients per year by preparing financial reports.In the preparation of financial reports,an intensive involvement of human resources from different levels is required.Due to the importance of rare and expensive human resources required in the preparation of financial reports,the management faces difficulties in the assignment of resources to the jobs.In other words management has to decide which jobs will be accepted and when they will be performed by utilizing resources at a maximum efficiency.

In order to make the problem more understandable, let us briefly describe how the jobs are handled by the audit firm:

In general,each audit is achieved in 3 stages,

1-Planning

2-Field

3-Reporting

The audit of year- end financial statements covers the majority of the work done by an audit firm during the year.The audit firm also performs audits at any time by the request of the client.

In the planning stage; engagement partner, engagement manager and previous year senior, budget the current year audit considering previous year experiences ; if the audit is the"first time through" type of an audit, the data of another firm in the same sector are considered for planning purposes.At the end of the planning stage for a job, expected partner, manager, senior and staff hours for the related audit work are determined. Within the firm, every level has a hourly per diem which is adjusted periodicaly. By multiplying expected hours with the related level's per diems and adding the level totals, the approximate price for the related job is calculated; and a proposal is made considering this price.When the audit work takes more than the expected(budgeted) completion time per client, the firm accumulates a loss in return of these excess work hours which were not charged to the client.

After a planning stage, proper audit teams and a schedule which indicates the beginning dates of the jobs are prepared.Audit work performed at the client's office or at the factory is called as "field work".Field work is completed in 2 stages as follows:

1)Interim

2)Final

For interim work there is no need for year-end financial statements of the client, so interim can be performed before the end of the year. After the client prepared and presented year-end financial statements, the audit team performs an audit on these financial statements beginning at a planned date after year end. At the end of the field work, the senior writes down the audit report under the supervision of engagement manager.After detailed controls and corrections of engagement manager and engagement partner the report is typed and signed by engagement partner and presented to the client.

Current Application of Scheduling within the Firm:

The firm makes proposals indicating prices to all previous year clients and potential first time through clients during the summer season just before the intensive work period which begins on September.Considering clients' requests such as due dates ,the jobs are scheduled manually for a two month period.At the end of each period, the scheduling is adjusted in order to allow the uncompleted jobs to continue and new jobs to begin such that they would be completed by their due dates.Since the whole year audit period is not considered in advance and the scheduling is not achived in line with it, during some periods the human resources are not enough to complete the jobs by regular daily work hours, whereas in other periods idle human resources may exist.The management has periodically prepared chargebility reports.Chargebility of any employee within the firm is defined for a period as the total hours spent by the employee for clients (which could be charged to the client) over the total working hours (which should be paid to the employee) for that period.The management aims to reach maximum chargibility for every employee within the firm.

THE SOLUTION PROCEDURE

The Improved Model:

In the present system the firm accepts all jobs in advance and then tries to schedule the jobs through out the year.In the improved model job selection is done such that chargibility is maximized when the jobs are scheduled.In other words acceptance and scheduling stages are combined to reach maximum chargibility.Assumptions made for the model are as follows:

1)Since the firm earns the same margin for all jobs, we assume that all jobs have equal preferences.

2)Since interim work can easily be performed in one or two weeks, there is no human resource capacity problem for the interims.Therefore only final works are considered.

3)As the year-end financial statement audits comprise the majority of the work done, we consider only year end audits.Considering the fact that most of the year-end audits are required to be completed until June, the model schedules the jobs for a period of 5 months (20 weeks) extending from January to the end of May.

4)Data related to the times spent by senior, experienced assistant and assistant for each job are received in hours and converted into weeks assuming 40 hours reqular work per week.We also assume that all jobs would be completed with integer number of weeks.

5)Field work of each audit is performed by a team.The team consist of a senior,experienced assistants,and assistants.In the model assignments of the seniors have the priority.

6)Managers and partners are not included in the model since they are basically involved in supervisory work.Most of the operations and paper work are perfomed by the senior, experienced assistants and assistants.

7)The number of human resources at each level are kept constant throughout the whole audit period.

8)It is assumed that every audit would be completed within the budgeted time limits.

Problem Formulation:

A linear programming (LP) model is constructed for the human resouce scheduling of an audit audit firm.Each audit is performed by a team of engagement partner, engagement manager,senior,experienced assistant and assistant.Since engagement manager and partner are not included in scheduling, every job is divided into 3 groups such as senior works, experienced assistant works and assistant works.

The decision variables of the model are as follows:

$$S_{i,j} = \begin{cases} 1 & \text{if i'th job is performed by a senior during j'th week} \\ 0 & \text{otherwise} \end{cases}$$

$$E_{i,j} = \begin{cases} 1 & \text{if i'th job is performed by an experienced assistant during j'th week} \\ 0 & \text{otherwise} \end{cases}$$

$$A_{i,j} = \begin{cases} 1 & \text{if i'th job is performed by an assistant during j'th week} \\ 0 & \text{otherwise} \end{cases}$$

where,

$i = 1,100$

$j = 1,20$

Constraints:

There are basically 5 group of restrictions in the model:

1-)Human resource capacity restrictions

2-)Audit completion time restrictions

3-)Structural restrictions

Human resource capacity and audit completion time restrictions are considered as explicit constraints of a binary integer programming problem.The structural restrictions are considered by assigning proper values to these variables from a given data set.

1)Human resource is the most important asset of an audit firm as the firm requires service of its personnel in the preparation of financial reports. Considering the fact that the audit firm has limited human resources at each level,the total number of jobs performed by each level are limited due to the capacity constraints.For example,the total number of senior works performed at week j can not be more than the total number of seniors at week j.Due to the assumption that the number of resources at each level are kept constant during the audit period,the resources that will be allocated to the jobs should be less than the related level's upper bound at any week.

So the human resource capacity constraint for each level can be stated as follows:

For the senior level;

$$\sum_{i=1,100} SAi * Si,j \leqslant SBj \qquad j=1,20$$

For the experienced assistant level;

$$\sum_{i=1,100} EAi * Ei,j \leqslant EBj \qquad j=1,20$$

For the assistant level;

$$\sum_{i=1,100} AAi * Ai,j \leqslant ABj \qquad j=1,20$$

where SAi,EAi,AAi denote number of resources required for the i'th job from senior, experienced assistant and assistant levels respectively. SBj,EBj,ABj denote capacities for senior,experienced assistant and assistant levels respectively.

2)As it is expected that every audit would be completed in a certain amount of time based on previous experiences,the following constraints are included in the model:

For the senior level;

$$\sum_{j=1,20} S_{i,j} = SC_i \qquad i=1,100$$

For the experienced assistant level;

$$\sum_{j=1,20} E_{i,j} = EC_i \qquad i=1,100$$

For the assistant level;

$$\sum_{j=1,20} A_{i,j} = AC_i \qquad i=1,100$$

where SCi,ECi,ACi denote the budgeted number of weeks for the i'th job for senior,experienced assistant and assistant works respectively.

3)Every audit work is performed by a team of senior,experienced assistants and assistants in consecutive weeks.Audit team begins field work together and experienced assistant and/or assistant may complete their works before senior as it is already planned.Due to the reasons such as Central Bank or Capital Market Board requirements, or companies' annual meetings, etc., each audit report should be completed up to a specific date requested by the client.These team work, consecutive weeks and due date requirements are satisfied using a data set instead of explicit constraint functions in the model.Data set includes data related to number of weeks which should be spent by each level for each job and due dates of each job.For example if the i'th job is planned to be completed in 4 senior weeks, 3 experienced assistant weeks and 2 assistant weeks,and if i'th job is planned to begin in in week j, the program checks whether it is suitable for the i'th job to begin in week j in order to be completed until the due date.If the answer is yes, the following variables are raised to 1 by retrieving the related information from the data set:

$S_{i,j} = 1$	$E_{i,j} = 1$	$A_{i,j} = 1$
$S_{i,j+1} = 1$	$E_{i,j+1} = 1$	$A_{i,j+1} = 1$
$S_{i,j+2} = 1$	$E_{i,j+2} = 1$	
$S_{i,j+3} = 1$		

Objective Function:

As the chargebility is the most important point that concerns management it is aimed to increase chargebility by maximizing the number of jobs which can be performed within a 20 week period.So our objective function will be as follows:

$$\text{Max } Z = \sum_{i=1,100} \sum_{j=1,20} S_{i,j} + \sum_{i=1,100} \sum_{j=1,20} E_{i,j} + \sum_{i=1,100} \sum_{j=1,20} A_{i,j}$$

In summary,the objective is to scehedule as many jobs as possible without violating constraints of the problem.

SOLVING ZERO-ONE LINEAR PROGRAMMING PROBLEMS USING IMPLICIT ENUMERATION

Review of Implicit Enumeration:

Implicit enumeration refers to a class of branch and bound algorithms designed specially for problems with binary decision variables.An exhaustive search of all possible solutions in a n variable 0-1 LP problem requires the enumeration of 2^n possible solutions.This is almost an impossible task to carry out in large problems.Implicit enumeration algorithm overcomes this problem by considering only some of the 2^n possible combinations, excluding the others which are concluded to be infeasible and/or suboptimal.These exclusions within the selected search employed, are conducted according to certain decision rules and tests.Although several refinements have been done by various authors,the original version of the implicit enumeration algorithm is credited to Balas (1). This original implicit enumeration algorithm developed by Balas(1) is inefficient in solving problems where the number of variables exceeds 30. This is largely because of the fact that Balas'(1) procedure has extensive bookkeeping requirements involving a large storage space and tedious data handling routines.Glover(3) has proposed some exclusion tests and applied the "backtracking" concept to implicit enumeration. Geoffrion (2) reformulated the Balas'(1) algorithm by representing the tree in vector form, which greatly improved the bookkeeping and computational efficiency of the algorithm.In this formulation, the path Pk from the initial node of the tree to node k is stored as a vector and uniquely determines the remaining enumeration required.Variable indices appear in the vector if they correspond to assigned variables.The order of the indices represents the level in the tree.A positive subscript indicates that the variable has been assigned the value one, and a negative subscript indicates that the variable has been assigned the value zero. In backtracking, the rightmost positive element is changed to a negative number and all negative elements to the right of it are dropped.The enumeration is complete when all remainig elements are negative.Petersen(6) has applied the implicit enumeration algorithm of Balas(1) with the modifications of Geoffrian(2) and Glover(3), to the selection of research and development projects. Tuan(7) has worked out a more flexible backtracking procedure in the tree search. Finaly Plane and McMillan(8) applied Geoffrion's(2) algorithm for solving zero-one problems considering exclusion tests developed by Fleischmann(5),Glover(3), Glover and Zionts(4) and Petersen(6).It is accepted that implicit enumeration is a powerful technique for zero-one problems due to the fact that it only employs additions and subtractions and these operations can be done very efficiently by a computer. Moreover,the technique does not require slack or surplus variables as in the simplex algorithm and the number of constraints do not grow as in the other integer programming algorithms.Finally, since the method is free of rounding errors, less storage space is needed in the solution of a given problem.

Concepts:

In order to fully understand the process, it is first necessary to define the following concepts;

Partial solution:In the implicit enumeration tree each node represents a "partial solution" where a subset of decision variables is assigned a value, either zero or one.

Free variables:Variables which are not included in the partial solution are

are called "free variables".

Completion of the partial solution: A solution formed by assigning binary values to all of the free variables is called a "completion" of the partial solution.In an n variable 0-1 LP problem, if a partial solution contains s variable, there are 2^{n-s} possible completions.

Fathoming a partial solution:In the implicit enumeration procedure for solving 0-1 LP problems, if a partial solution cannot be completed in such a way as to avoid violating one or more of the constraints or if it becomes evident that all possible completions of the present partial solution cannot improve the objective function value it is said that,that partial solution has been "fathomed", meaning that all completions of that partial solution are implicitly enumerated and can be ignored.

Backtracking:When a partial solution is fathomed, in order to examine other sets of solutions which have not yet been enumerated, a form of backtracking is required.Backtracking should be done such that we do not miss any of the possible combinations while making sure that we do not reevaluate any of the solutions we have implicitly enumerated before.

Standart Form of the Zero-One LP Problem:

The implicit enumeration algorithm developed by Plane and McMillan(8) requires the 0-1 LP problem to be formulated as follows:

1) All constraints are expressed as a " $\geqslant$ " constraint.

2) All coefficients in the objective function are positive or zero.

3) The value of the objective function is aimed to be minimized.

The first attribute enables us to identify quickly which variables, in a violated constraint, could help bring about feasibility by being raised to 1.

The second and third attributes enable us to deduce that no completion of a feasible partial which required raising additional variables to 1 would be attractive.

Reformulating the constraints in conformance with 1 above is easy.We simply put all elements to the left of the inequality sign(including the constant, if any).If necessary we then multiply by -1 to change $\leqslant 0$ to $\geqslant 0$.If we have equality constraints we replace them with inequality constraints as follows:

Given: $g1 = c1$

$g2 = c2$
: :
: :
: :
$gn = cn$

Replace: $g1 \geqslant c1$

$g2 \geqslant c2$
: :
: :
: :
$gn \geqslant cn$

and add the following constraint,

$$g1 + g2 + \ldots\ldots\ldots\ldots + gn \leqslant c1 + c2 + \ldots\ldots\ldots\ldots cn$$

Note that with this formulation the number of constraints is only increased by 1.

Reformulating an objective function where necessary to cause all coefficients to be possitive or zero (2) is also straight forward.We simply replace each variable (Xj) which has a negative coefficient (Cj) in the objective function with a new variable which is one less the original variable. For example, if Xj has a negative Cj, a new variable Yj is defined such that Yj = 1 - Xj . Substitution of 1 - Yj for Xj will convert Cj to a positive value. This substitution should be realized in the constraints as well as the objective function of the problem. If the objective function is aimed to be maximized, we multiply the objective function by -1 and then proceed to minimize it.Any negative coefficient which this multiplication produces is transformed as suggested above.

The maximization objective function is multiplied by -1 to convert the problem into a minimization problem as a result of which all objective function coefficients became negative.This is corrected by proper substitution of the variables with new ones.Finally the " $\leqslant$ " and " = " constraints of the problem were converted to " $\geqslant$ " constraints as suggested by Plane and McMillan.The standartized form of the objective function and constraints are as follows:

Objective Function:

$$\text{Min } Z = \sum_{i=1,100} \sum_{j=1,20} NSi,j + \sum_{i=1,100} \sum_{j=1,20} NEi,j + \sum_{i=1,100} \sum_{j=1,20} NAi,j$$

Constraints:

For the senior level;

$$\sum_{i=1,100} SAi * NSi,j \geqslant 100 - SBj \qquad j=1,20$$

$$\sum_{j=1,20} - NSi,j \geqslant SCi - 20 \qquad i=1,100$$

$$\sum_{i=1,100} \sum_{j=1,20} NSi,j \geqslant 2000 - \sum_{i=1,100} SCi$$

For the experienced assistant level;

$$\sum_{i=1,100} EAi * NEi,j \geqslant 100 - EBj \qquad j=1,20$$

$$\sum_{j=1,20} - NEi,j \geqslant ECi - 20 \qquad i=1,100$$

$$\sum_{i=1,100} \sum_{j=1,20} NEi,j \geqslant 2000 - \sum_{i=1,100} ECi$$

For the assistant level;

$$\sum_{i=1,100} AAi * NAi,j \geqslant 100 - ABj \qquad j=1,20$$

$$\sum_{j=1,20} - NAi,j \geqslant ACi - 20 \qquad i=1,100$$

$$\sum_{i=1,100} \sum_{j=1,20} NAi,j \geq 2000 - \sum_{i=1,100} ACi$$

where,

$NSi,j = 1\text{-}Si,j$

$NEi,j = 1\text{-}Ei,j$

$NAi,j = 1\text{-}Ai,j$

Reformulating the objective function and constraints in conformance with the algorithm's requirements we increased the total number of constraints by 3 because of the equality constraints.

Exclusion Tests and Methods:

Many tests for reducing the enumeration have been proposed.However, these tests can reduce the enumeration at the expense of added calculation, and they should be applied as long as the extra calculations do not offset the benefits of reduced enumeration.Balas(1) brought forward three criteria for exclusion in an attempt to improve efficiency of the algorithm.

Balas' first test:If for any constraint the infeasibility can not be diminished even though all the variables with negative coefficients are entered, then that node is fathomed.

Balas' second test:If for any free variable, assignment of value 1 will make the present value of the objective function value less than the lower bound so far computed, that node is fathomed.

Balas' third test:If for all violated constraints, the coefficients of the entering variable are positive,that node is fathomed since it can not help to achieve feasibility.

In a note following Balas'(1) paper, Glover(3) also suggested modifications to increase the efficiency of the algorithm.Some other tests were suggested by Geoffrian(2), Fleischmann(5), Glover(3), Glover and Zionts(4) and Petersen(6). Plane and McMillan(8) summarized these tests as two methods for implicitly enumerating solutions as follows:

1)A group of solutions is implicitly enumerated by infeasibility if it is not possible to satisfy every constraint by any completion of the current partial solution.

2)A group of solutions is implicitly enumerated by feasibility when the partial solution being considered is feasible when it is completed by setting all remaining variables equal to zero.It is known that it would never be attractive to set additional variables equal to one, because this would never decrease the value of the objective function.This is because all of the objective function coefficients are positive or reformulated as positive numbers.

The General Procedure Improved by Plane and McMillan for Solving Zero-One LP Problems Via Implicit Enumeration:

Before reviewing the procedure improved by Plane and McMillan(8), let us briefly describe the symbolic notation used for easier reference;

f :It represents the value of the objective function.

Sk :It represents the set of variables which have been given specific values in a partial solution at step k.The elements stored at Sk will be the subscripts of the variables which have been assigned specific values in the k'th partial solution.If a subscript has a minus sign in front of it this means that the variable which that subcript specifies has a value of 0 in that partial solution;otherwise it has a value of 1.The variables are also stored at Sk from left to right in the order in which they are assigned. It is also helpful to define a free variable as any variable which is not in Sk.

Vk :It represents the set of constraints which are violated when the partial solution is completed by setting to zero all free variables.

Tk :It represents the set of variables which are both free variables and have positive coefficients in one or more constraints in Vk.

X :It represents the best feasible solution thus far found-that is the feasible solution which thus far yields the smallest value for f.

Z :It represents the value of f when the objective function is evaulated at solution X.

The general procedure is portrayed in the flow diagram of Fig.1 .The procedure begins by setting all variables to zero in order to check whether the solution "all X's=0" is feasible.If it is so the problem is solved otherwise an upper bound is calculated by evaluating f at setting all variables to 1.After the determination of starting value for Z, the algorithm cycle begins with S empty.The algorithm identifies the set of violated constraints,V.It stores in T the variables which, by being raised to 1,might bring about feasibility.If it possible to make every constraint in V feasible by adding variables in T,the algorithm adds to S that variable in T which is most "helpful" as measured by the sum of its coefficients in all of the constraints.This process is continued until a partial solution is fathomed,either by feasibility or infeasibility. When backtracking is applied,the general systematic procedure will be to complement the rightmost element in S which has not yet been complemented; that is, the rightmost positive element in S is found and replaced with its negative complement.At this step it is necessary to drop any element in S to the right of the variable currently being complemented and being again to fathom the new partial solution.The procedure continues until all of the elements in S are negative. The incumbent solution, if any, is an optimal solution.If no incumbent solution has been found,there is no feasible solution better than the solution corresponding to the best upper bound which was already found.

The Modified Implicit Enumeration Algorithm:

We have modified the implicit enumeration algorithm improved by Plane and McMillan(8) in order to solve 0-1 integer programming problem formulated for the human resource scheduling in an audit firm.The final form of the algorithm is obtained by adding the following improvements to the previous procedure:

1-)Although the variables are processed one by one in the previous algorithm, the new procedure considers a group of variables since it is almost impossible to include these restrictions in the constraints. These restrictions are satisfied in advance using a data set instead of explicit constraint functions. The variable groups that satisfy the following restrictions are processed throughout the algorithm:

-- When a variable related to seniors is assigned, variables related to experienced assistants and assistants should be assigned since every audit is performed by a team of senior, experienced assistants and assistants.

-- When any variable related to seniors, experienced assistants and assistants is assigned, consecutive variables in the same categories should be assigned since every audit is performed in consecutive weeks.

-- Assignment of variables are restricted by due dates since every audit should be completed up to a specific date due to the reasons such as Central Bank and Capital Market Board requirements and companies' annual meetings.

Data set includes data related to number of weeks which should be spent by each level for each job and due dates of each job. Considering the above mentioned restrictions and retrieving the related information from the data data set, the program makes assignments for variable groups.

Variables related to seniors are considered as the representative of the variable groups and when any of these variables is assigned, the corresponding group of variables is assigned by the help of the data set.

2-) In the general enumeration algorithm, the variable with the greatest coefficient sum is added to the partial solution.In the modified algorithm, a variable group determined by the data set is added to the partial solution without any priority.

These modifications have simplified the model and resulted in a decrease in computational time.

ANALYSIS OF RESULTS

Application of the modified implicit enumeration algorithm to the integer programming problem yields the selection of jobs which will be performed during a period of 5 months extending from January to the end of May. Omitting overtime capacity it was possible to select and schedule 69 jobs out of 100 jobs performed in 1989. The optimum value of the objective function, which is the total number of assigned resources, is 730 men-weeks. Since the model is based on 1989 working period data, we can compare the number of jobs selected and chargibility obtained for each resource with actual values of 1989.

TABLE 1.
Summary of 1989 Performance

		Seniors	Exp.Assistants	Assistants
Total capacity for the period(hrs)		16,800	8,800	12,000
Chargable hours	Normal	14,670	6,230	9,310
	Overtime	7,250	2,930	3,810
Chargibility %		79%	69%	74%

Number of jobs performed with overtime = 100

TABLE 2.

Summary of Model Outputs Without Overtime

		Seniors	Exp.Assistants	Assistants
Total capacity for the period(hrs)		16,800	8,800	12,000
Chargable hours	Normal	15,320	5,460	8,840
Chargibility %		91%	64%	74%

Number of jobs scheduled without overtime = 69

As can be seen from a comparison of TABLE 1. and TABLE 2.,chargibility percentage for seniors increased from 79% to 91% while chargibility percentage for experienced assistants decreased from 69% to 64% and chargibility percentage for assistants remained constant.

These results are highly satisfactory since the cost of human resources for the senior category is much more higher than the other levels and overtime is not desired by the management due to the bid price effect in such a competitive market.

On the other hand, in order to achieve an accurate comparison of model outputs with actual results, the program was run by increasing capacities as much as overtimes realized in 1989 and the results were presented in TABLE 3. as follows:

TABLE 3.

Summary of Model Outputs With Overtime

		Seniors	Exp.Assistants	Assista
Total capacity for the period(hrs)		16,800	8,800	12,00
Chargable hours	Normal	16,760	7,880	11,320
	Overtime	5,120	1,280	1,720
Chargibility %		89%	85%	89%

Number of jobs scheduled with overtime = 100

As can be seen from a comparison of TABLE 1. and TABLE 3., not only the chargibility percentages were increased but also the required overtime hours were decreased by considerable amounts. The chargibility percentage for seniors increased from 79% to 91% , the chargibility percentage for experienced assistants increased from 69% to 85% and the chargibilty percentage for assistants increased from 74% to 89% . Since the model scheduled the jobs more efficiently throughout the working period, less overtime is required in order to complete all jobs.

CONCLUSION

The application of a linear programming model to an audit firm yields an overall human resource scheduling over a planning period such that the chargibility or the utilization of resources is maximized.

The determination of jobs and scheduling of human resources in an audit firm is achieved with 2 major contributions as follows:

1-) We have formulated a model that combines selection and scheduling stages. In the model, job selection is done such that chargibility is maximized when the jobs are scheduled.

2-) We have modified the implicit enumeration algorithm developed by Plane and McMillan. In order to simplify the problem and to decrease computational time, the modified algorithm processes variables in groups instead of considering them individualy.

Computational experience resulted with the decision of which jobs will be accepted and when they will be performed by utilizing resources at a maximum efficieny.

The model helps the management to decide which jobs to accept and advises a schedule that will utilize resources with maximum efficiency.

The application of the model does not induce any significant cost to the firm. Because of the simplicity of application ,once the datas are entered the application can easily be performed.

APPENDIX

Definition of Variables Used in the Program:

SS(I,J) :Partial solution matrix to keep track whether the corresponding variable related to seniors is free(=0),assigned one(=11) or assigned zero (=10)

EE(I,J) :Partial solution matrix to keep track whether the corresponding variable related to experienced assistants is free(=0),assigned one(=11) or assigned zero (=10)

AA(I,J) :Partial solution matrix to keep track whether the corresponding variable related to assistants is free(=0),assigned one(=11) or assigned zero (=10)

S(I,J) :Matrix containing the value of the corresponding SS(I,J).If SS(I,J) is assigned to 11, S(I,J) will be assigned to 1 otherwise to 0.

E(I,J) :Matrix containing the value of the corresponding EE(I,J).If EE(I,J) is assigned to 11, E(I,J) will be assigned to 1 otherwise to 0.

A(I,J) :Matrix containing the value of the corresponding AA(I,J).If AA(I,J) is assigned to 11, A(I,J) will be assigned to 1 otherwise to 0.

SAA(I,J):Binary problem constraint coefficients matrix for senior category.

EAA(I,J):Binary problem constraint coefficients matrix for experienced assistant category.

AAA(I,J):Binary problem constraint coefficients matrix for assistant category.

SB(I) :Binary problem right hand side array for senior category.

EB(I) :Binary problem right hand side array for experienced assistant category.

AB(I) :Binary problem right hand side array for assistant category.

CS(I,J) :Binary problem objective function coefficients matrix for senior category.

CE(I,J) :Binary problem objective function coefficients matrix for experienced assistant category.

CA(I,J) :Binary problem objective function coefficients matrix for assistant category.

VS(I,J) :Matrix containing the values when the partial solution is completed by setting to zero all free variables.If SS(I,J) is assigned to zero, VS(I,J) will be assigned to 0 otherwise to S(I,J).

VE(I,J) :Matrix containing the values when the partial solution is completed by setting to zero all free variables.If EE(I,J) is assigned to zero, VE(I,J) will be assigned to 0 otherwise to E(I,J).

VA(I,J) :Matrix containing the values when the partial solution is completed by setting to zero all free variables.If AA(I,J) is assigned to zero, VA(I,J) will be assigned to 0 otherwise to A(I,J).

SD(I) :Array containing the value of the left hand side of each constraint evaluated at VS(I,J).

ED(I) :Array containing the value of the left hand side of each constraint evaluated at VE(I,J).

AD(I) :Array containing the value of the left hand side of each constraint evaluated at VA(I,J).

SDD(I) :Array of flags indicating whether the corresponding constraint related to seniors is violated(=1) or not(=0).

EDD(I) :Array of flags indicating whether the corresponding constraint related to experienced assistants is violated(=1) or not(=0).

ADD(I) :Array of flags indicating whether the corresponding constraint related to assistants is violated(=1) or not(=0).

TDD :Number of violated constraints.

ZI :Value of objective function evaluated at PS(I,J),PE(I,J),PA(I,J)

FP :Value of objective function evaluated at VS(I,J),VE(I,J),VA(I,J)

Z :Limit for variable selection.

STT(I) :Array of flags indicating which constraints should be considered for T storages for senior category.

ETT(I) :Array of flags indicating which constraints should be considered for T storages for experienced assistant category.

ATT(I) :Array of flags indicating which constraints should be considered for T storages for assistant category.

TTS(I,J):Matrix containing the values when partial solution is completed by setting to 1 all free variables.If SS(I,J) is assigned to 0,TTS(I,J) will be assigned to 1 otherwise to S(I,J).

TTE(I,J):Matrix containing the values when partial solution is completed by setting to 1 all free variables.If EE(I,J) is assigned to 0,TTE(I,J) will be assigned to 1 otherwise to E(I,J).

TTA(I,J):Matrix containing the values when partial solution is completed by setting to 1 all free variables.If AA(I,J) is assigned to 0,TTA(I,J) will be assigned to 1 otherwise to A(I,J).

ST(I) :Array containing the value of the left hand side of each constraint evaluated at TTS(I,J).

ET(I) :Array containing the value of the left hand side of each constraint evaluated at TTE(I,J).

AT(I) :Array containing the value of the left hand side of each constraint evaluated at TTA(I,J).

TS(I) :Array to keep track whether the corresponding constraint for senior category have any variable or not.

TE(I) :Array to keep track whether the corresponding constraint for experienced assistant category have any variable or not.

TA(I) :Array to keep track whether the corresponding constraint for assistant category have any variable or not.

PS(I,J) :Matrix containing the best feasible solution thus far found for variables related to seniors.

PE(I,J) :Matrix containing the best feasible solution thus far found for variables related to experienced assistants.

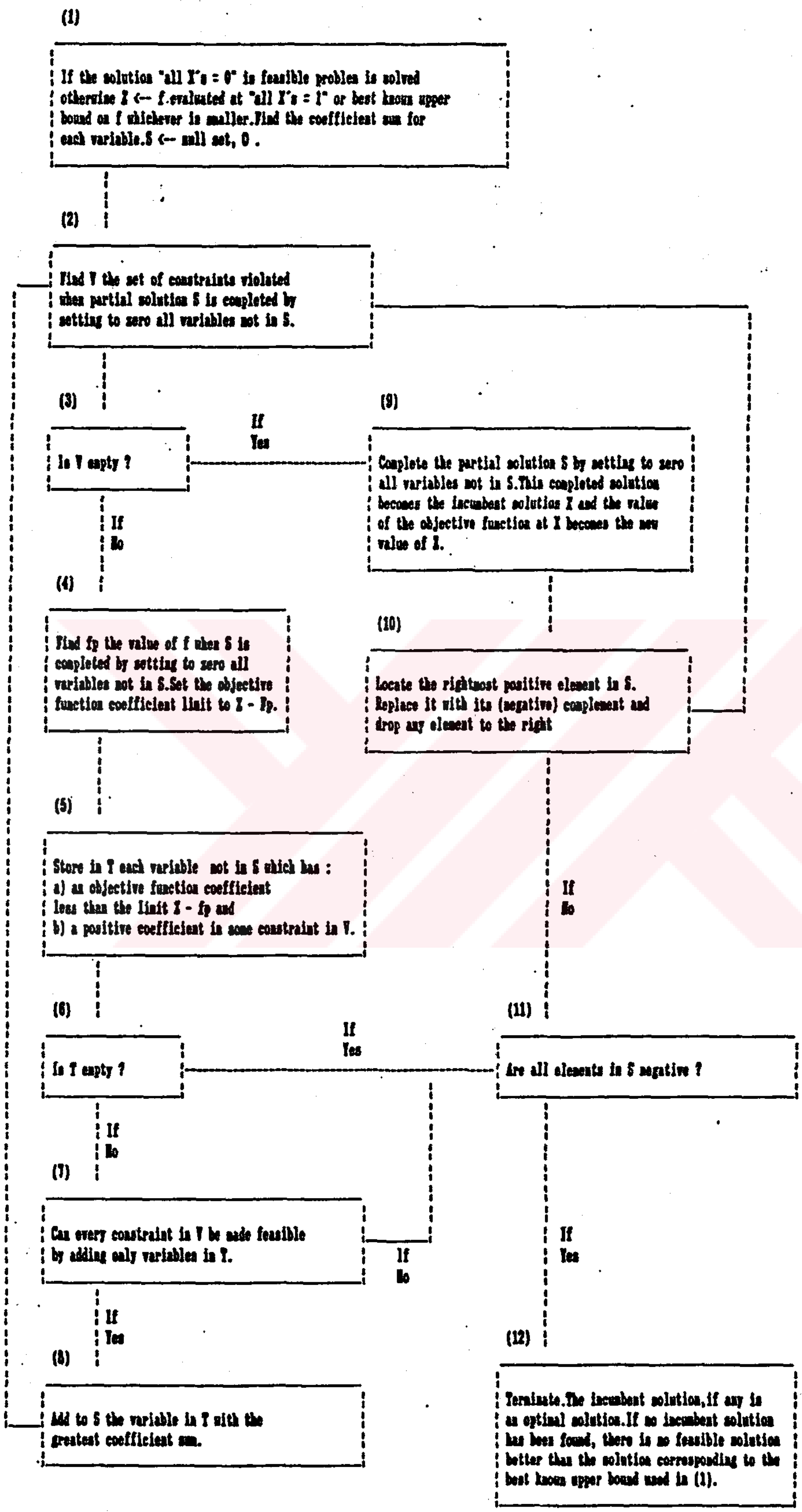
PA(I,J) :Matrix containing the best feasible solution thus far found for variables related to assistants.

DAT(I,J):Data matrix containing audit times per each employee category for each job.

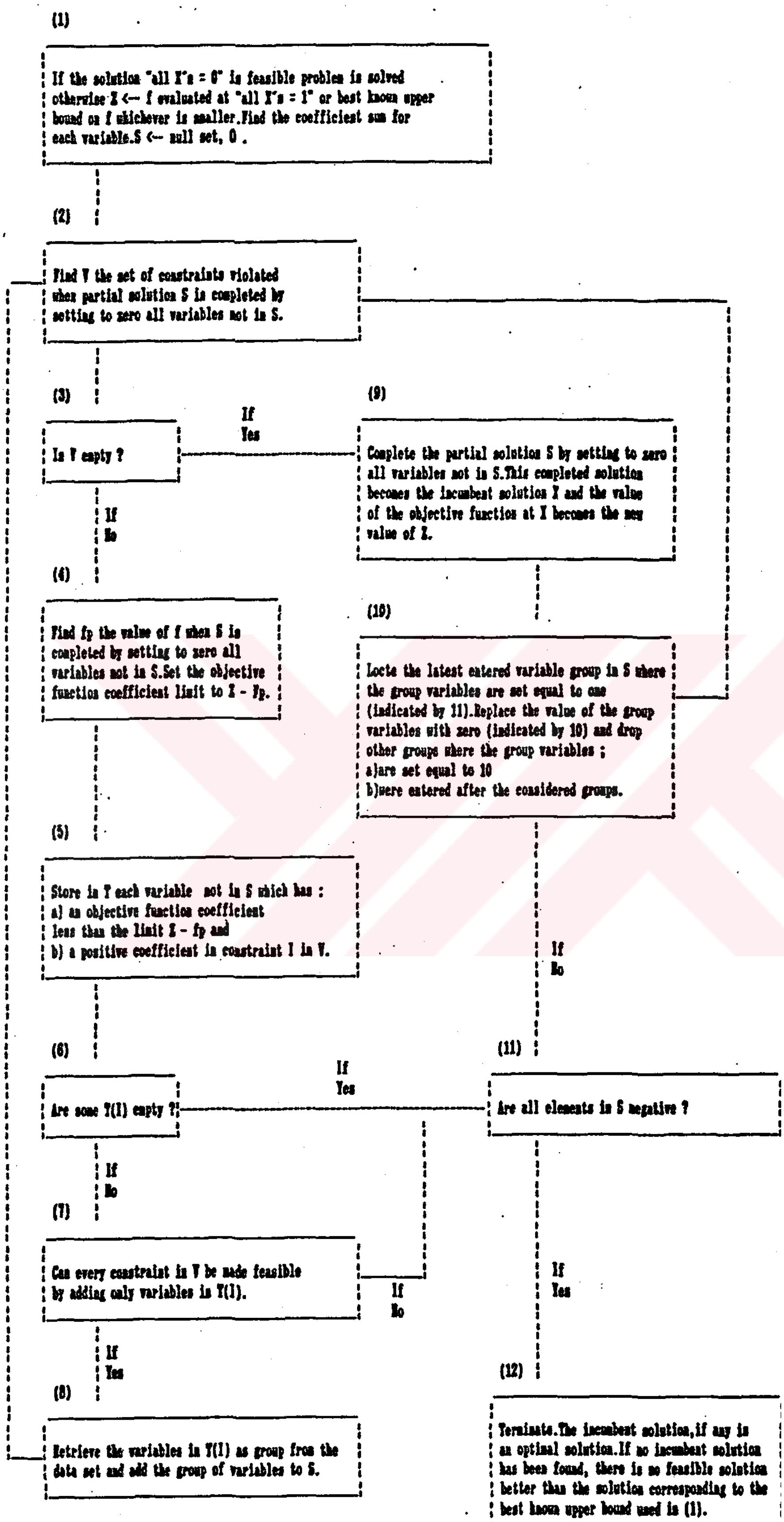
MAX(I) :Data array indicating maximum starting week of each job.

K :Clock to keep track iteration numbers.

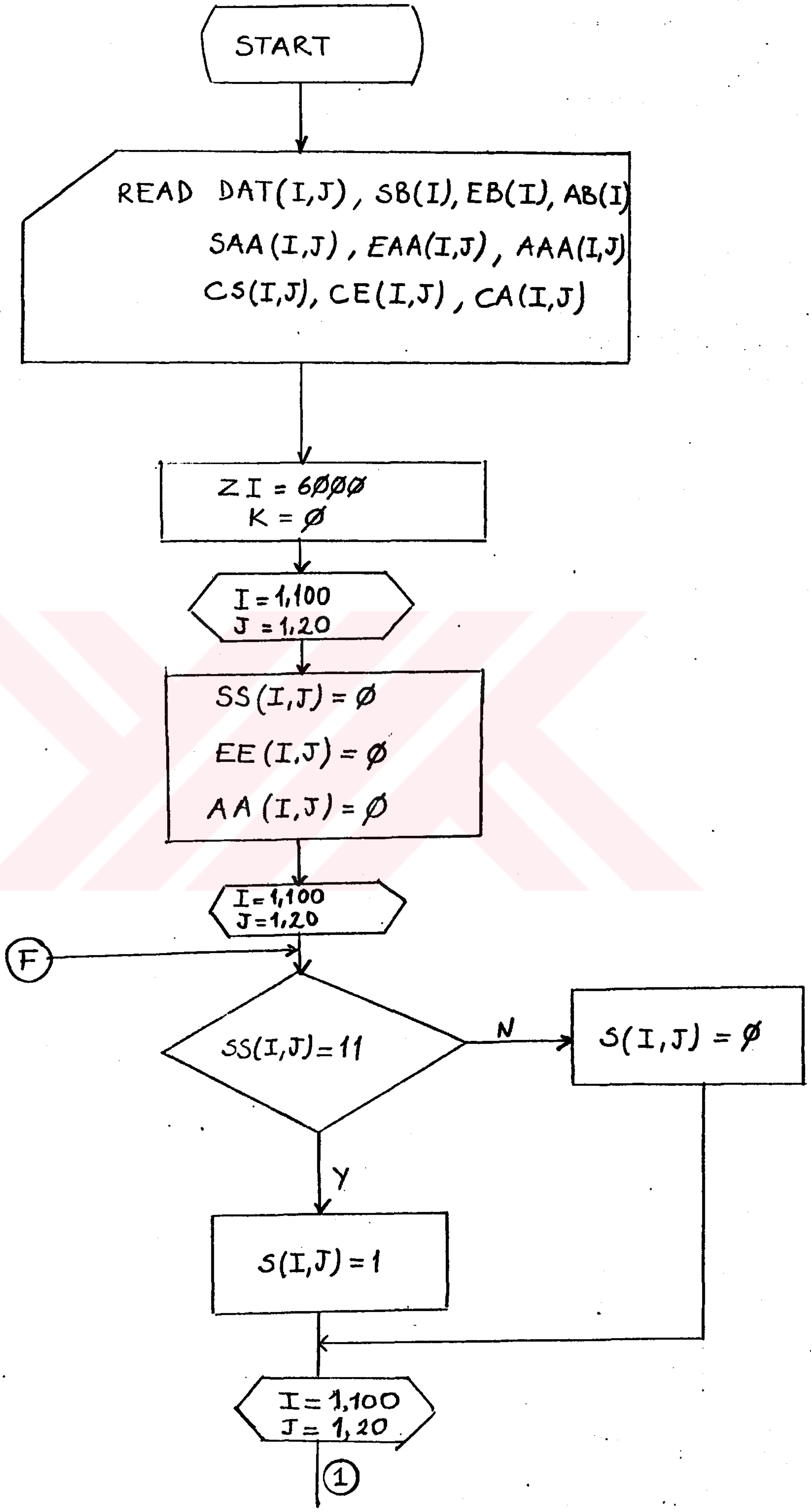
GENERAL FLOWCHART OF IMPLICIT ENUMERATION ALGORITHM

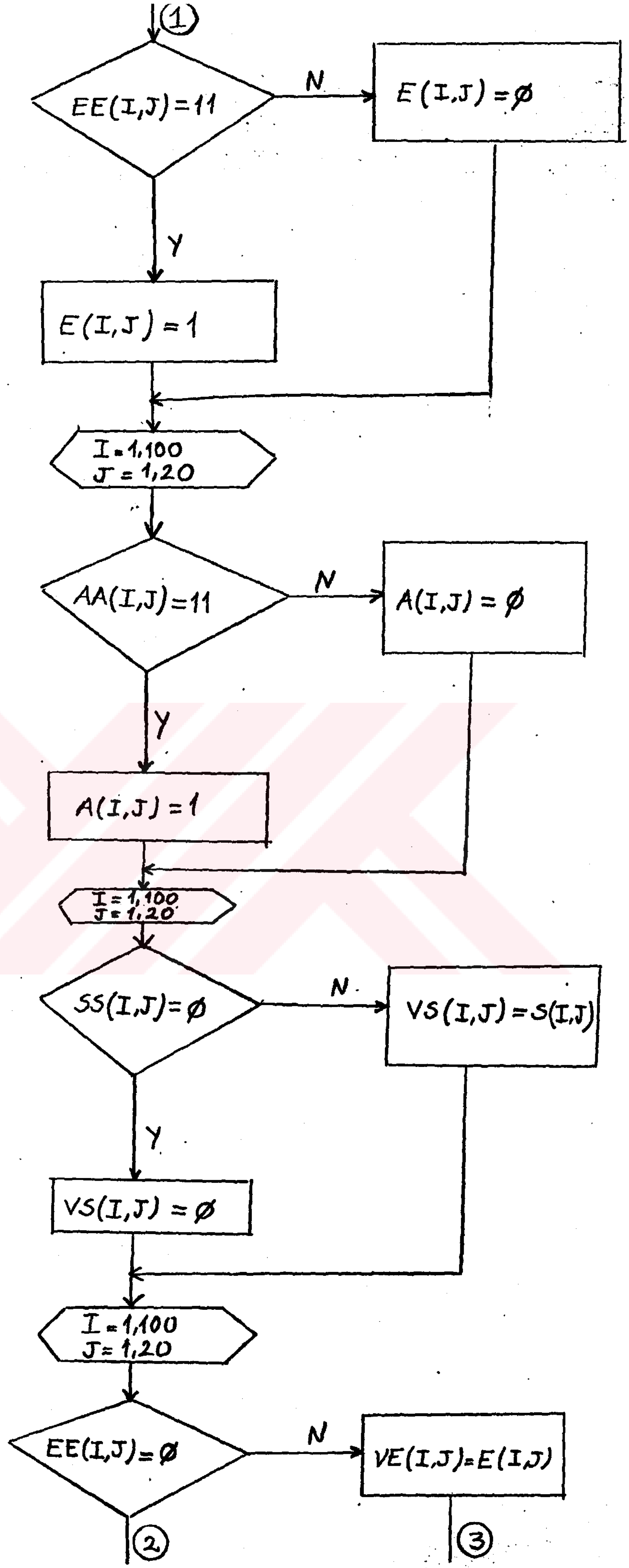


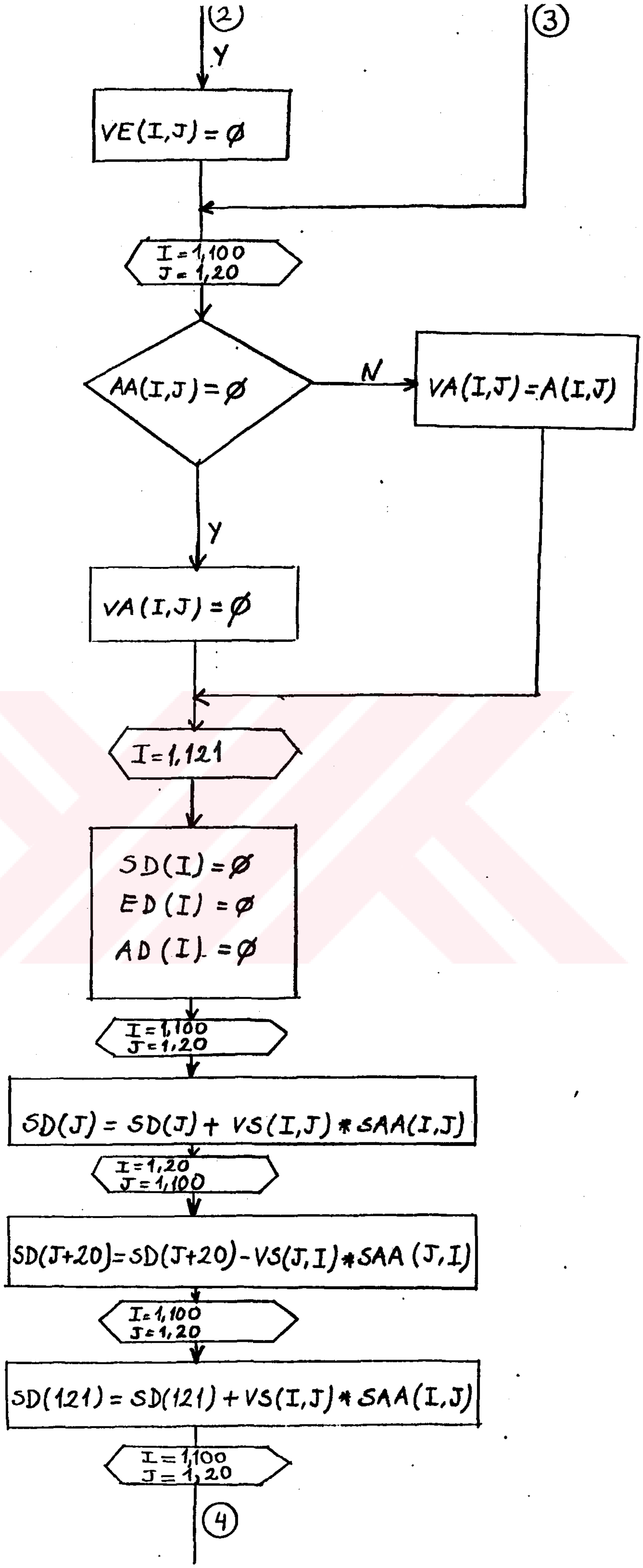
FLOWCHART OF MODIFIED IMPLICIT ENUMERATION ALGORITHM

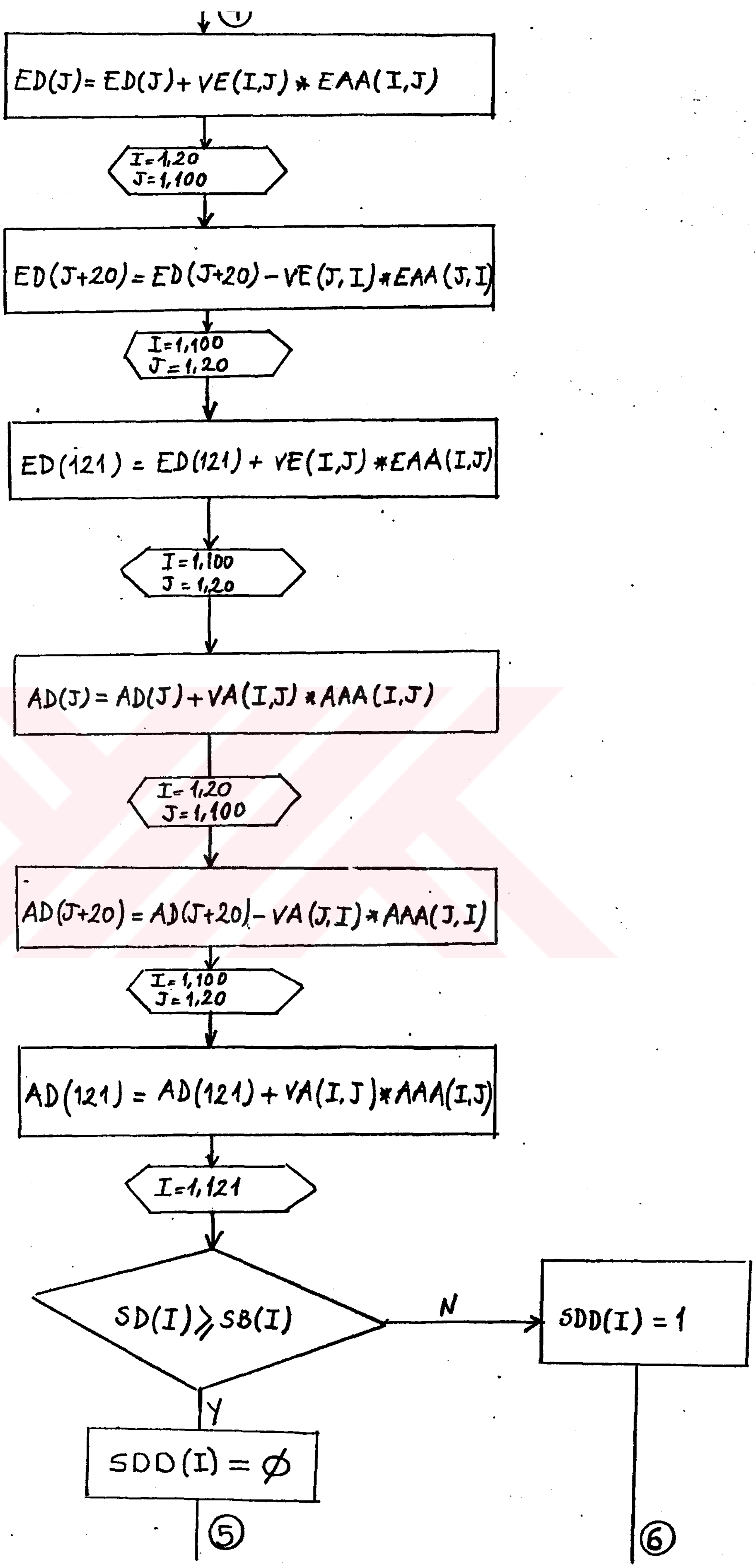


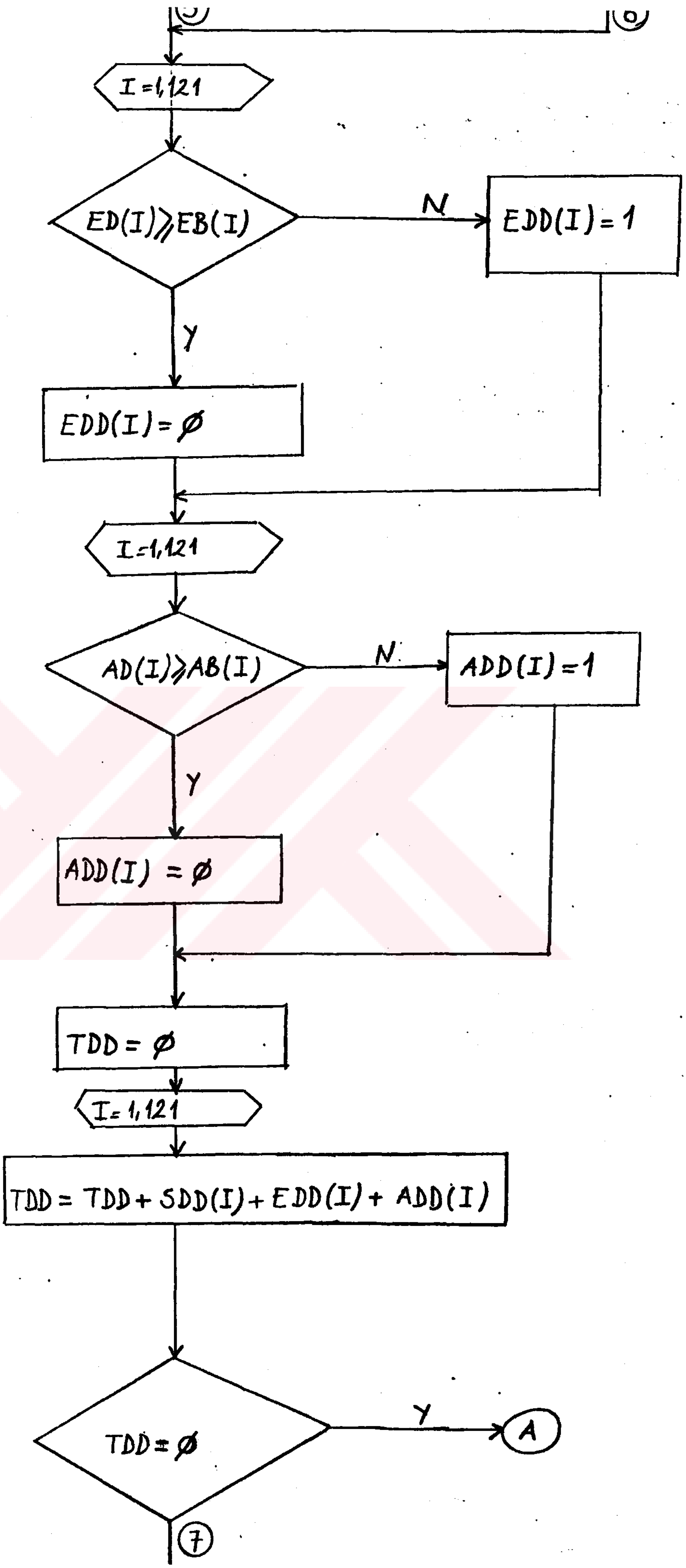
COMPUTER FLOWCHART OF THE MODIFIED ALGORITHM



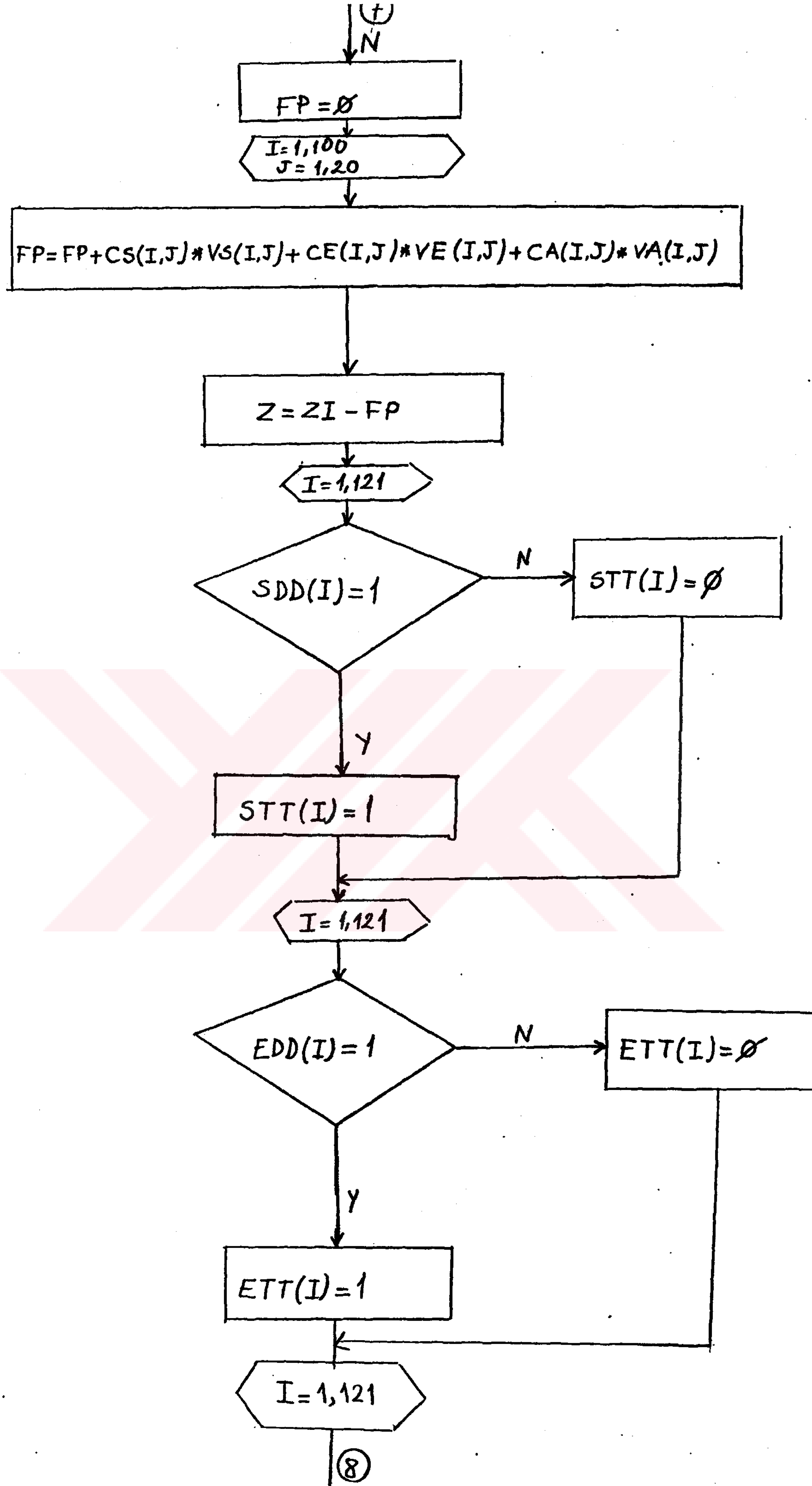

1
EE(I,J)=11
N
E(I,J)=Ø
Y
E(I,J)=1
I=1,100
J=1,20
AA(I,J)=11
N
A(I,J)=Ø
Y
A(I,J)=1
I=1,100
J=1,20
SS(I,J)=Ø
N
VS(I,J)=S(I,J)
Y
VS(I,J)=Ø
I=1,100
J=1,20
EE(I,J)=Ø
N
VE(I,J)=E(I,J)
2
3


2
3
Y
VE(I,J) = ∅
I = 1,100
J = 1,20
AA(I,J) = ∅
N
VA(I,J) = A(I,J)
Y
VA(I,J) = ∅
I = 1,121
SD(I) = ∅
ED(I) = ∅
AD(I) = ∅
I = 1,100
J = 1,20
SD(J) = SD(J) + VS(I,J) * SAA(I,J)
I = 1,20
J = 1,100
SD(J+20) = SD(J+20) - VS(J,I) * SAA(J,I)
I = 1,100
J = 1,20
SD(121) = SD(121) + VS(I,J) * SAA(I,J)
I = 1,100
J = 1,20
4

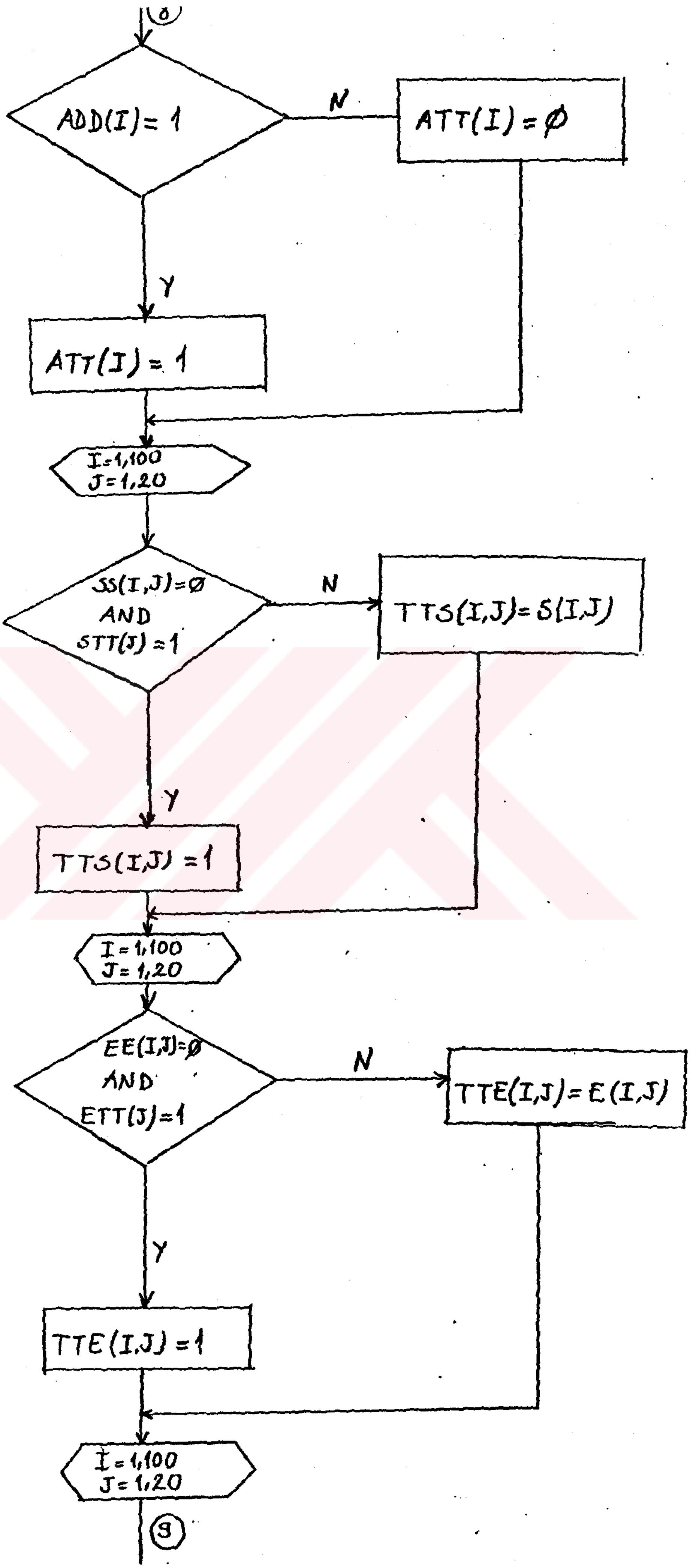

4
ED(J)= ED(J)+ VE(I,J) * EAA(I,J)
I=1,20
J=1,100
ED(J+20) = ED(J+20) - VE(J,I) * EAA(J,I)
I=1,100
J=1,20
ED(121) = ED(121) + VE(I,J) * EAA(I,J)
I=1,100
J=1,20
AD(J) = AD(J) + VA(I,J) * AAA(I,J)
I=1,20
J=1,100
AD(J+20) = AD(J+20) - VA(J,I) * AAA(J,I)
I=1,100
J=1,20
AD(121) = AD(121) + VA(I,J) * AAA(I,J)
I=1,121
SD(I) ≥ SB(I)
N
SDD(I) = 1
Y
SDD(I) = ∅
5
6

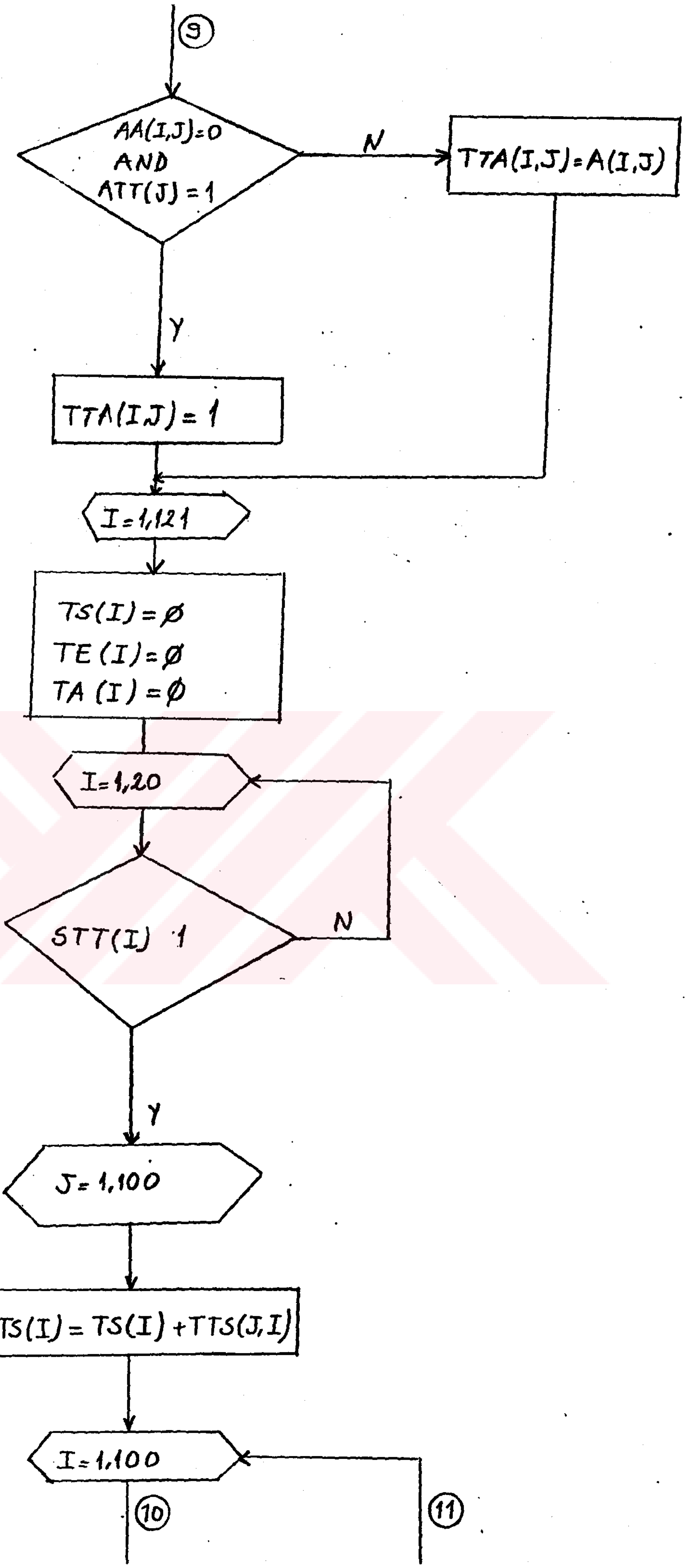


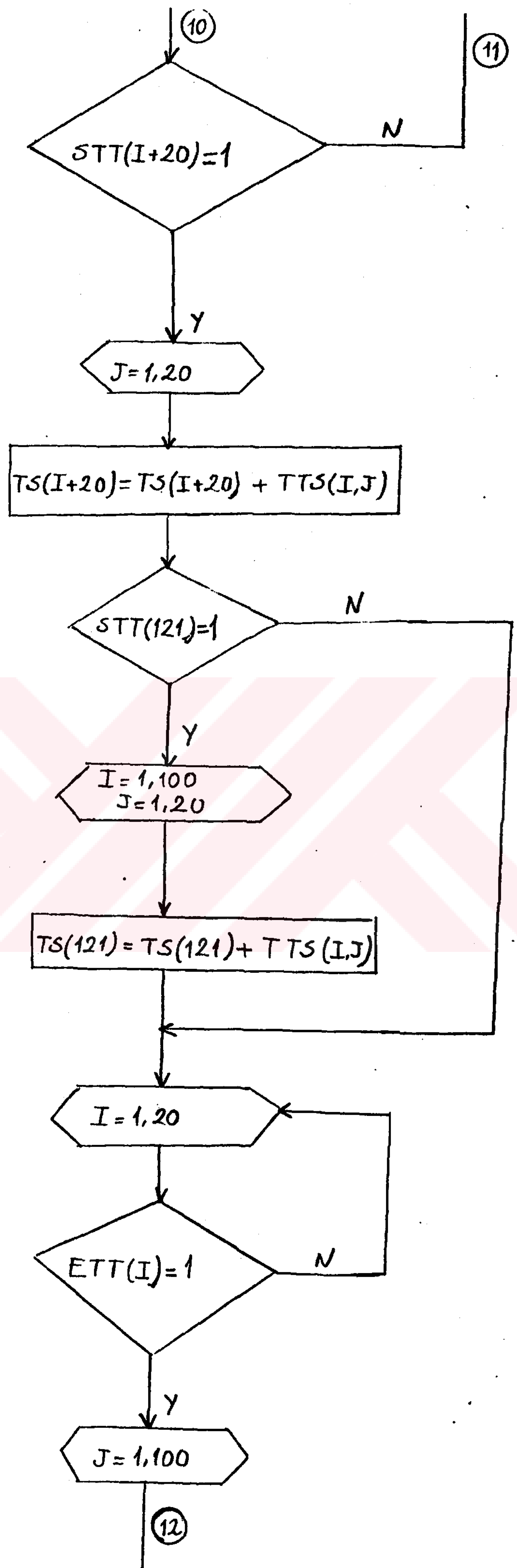
5
6
I = 1,121
ED(I) ≥ EB(I)
N
EDD(I) = 1
Y
EDD(I) = Ø
I = 1,121
AD(I) ≥ AB(I)
N
ADD(I) = 1
Y
ADD(I) = Ø
TDD = Ø
I = 1,121
TDD = TDD + SDD(I) + EDD(I) + ADD(I)
TDD = Ø
Y
A
7



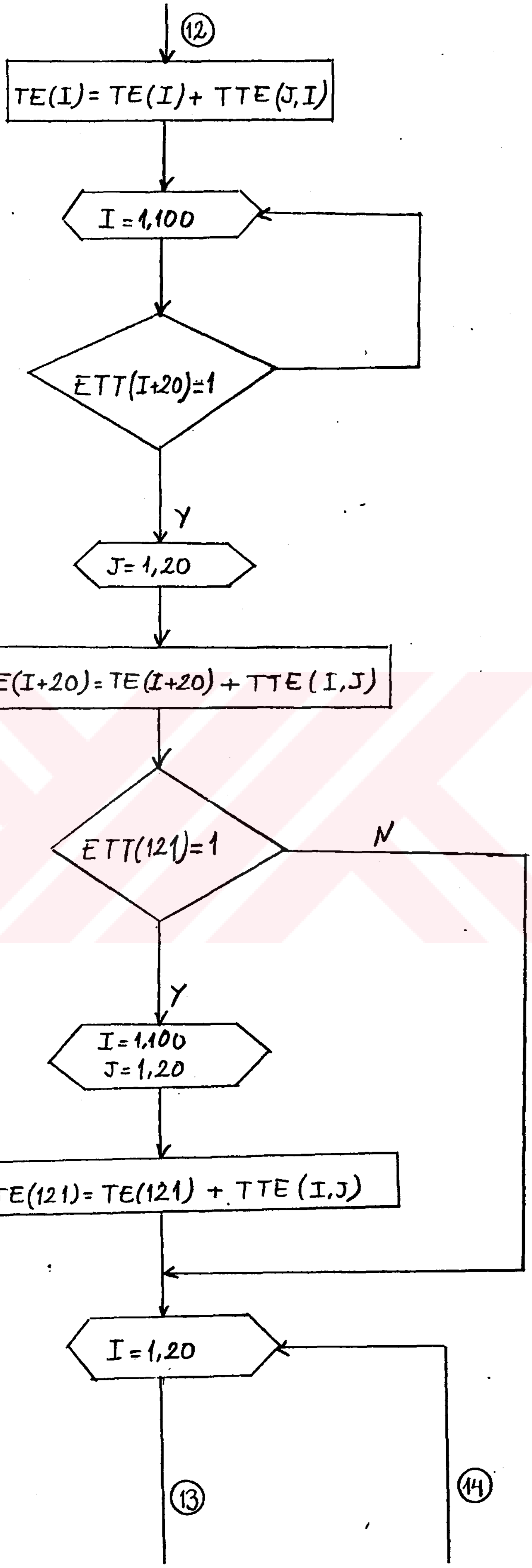
(7)
N
FP = Ø
I = 1,100
J = 1,20
FP = FP + CS(I,J) * VS(I,J) + CE(I,J) * VE(I,J) + CA(I,J) * VA(I,J)
Z = ZI − FP
I = 1,121
SDD(I) = 1
N
STT(I) = Ø
Y
STT(I) = 1
I = 1,121
EDD(I) = 1
N
ETT(I) = Ø
Y
ETT(I) = 1
I = 1,121
(8)

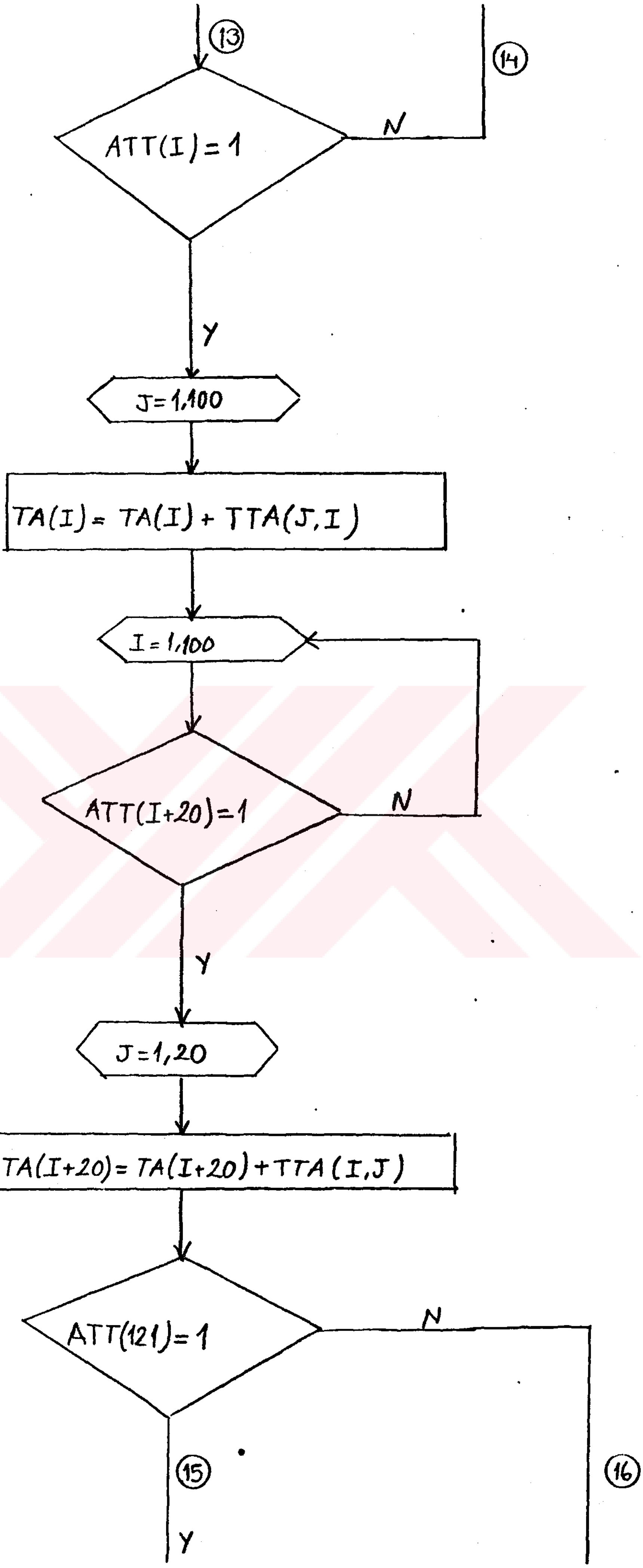

8
ADD(I) = 1
N
ATT(I) = Ø
Y
ATT(I) = 1
I = 1,100
J = 1,20
SS(I,J) = Ø
AND
STT(J) = 1
N
TTS(I,J) = S(I,J)
Y
TTS(I,J) = 1
I = 1,100
J = 1,20
EE(I,J) = Ø
AND
ETT(J) = 1
N
TTE(I,J) = E(I,J)
Y
TTE(I,J) = 1
I = 1,100
J = 1,20
9

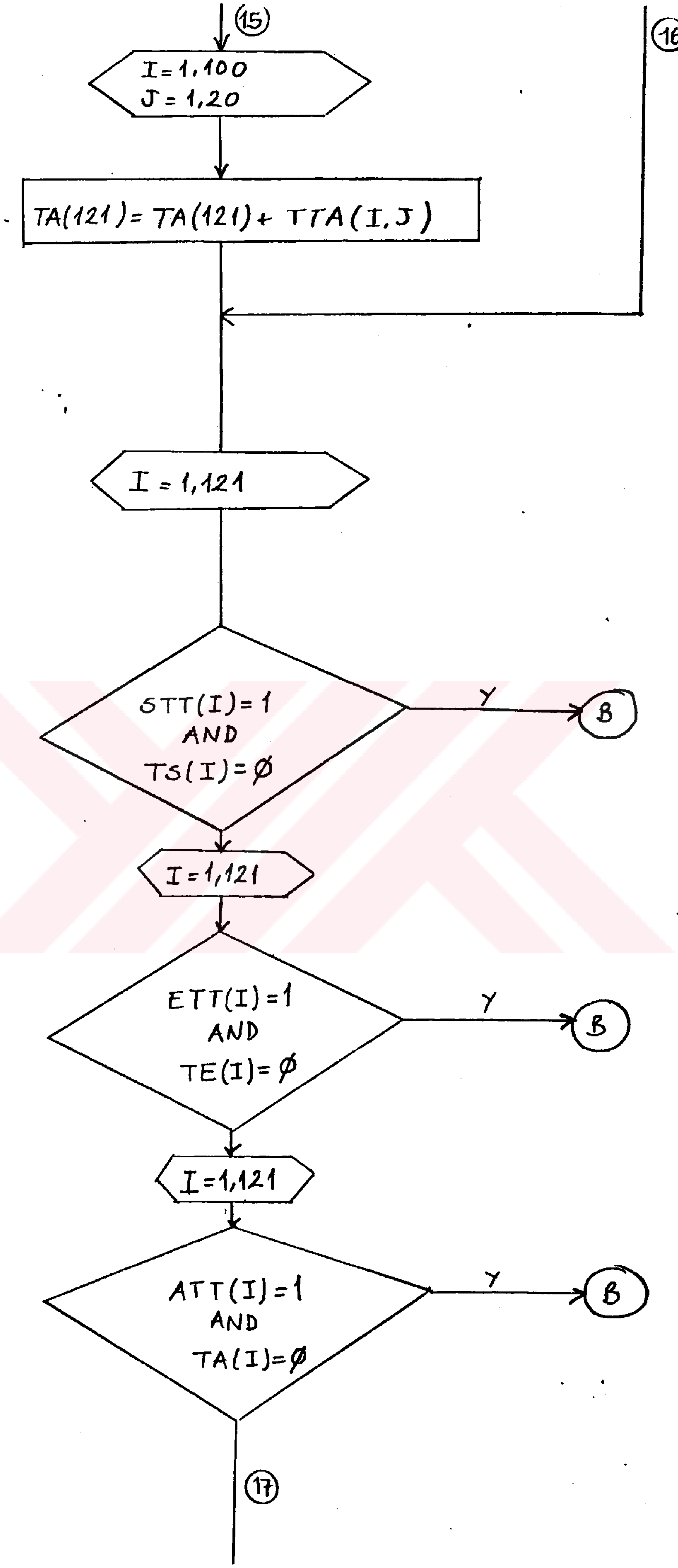

9
AA(I,J)=0
AND
ATT(J)=1
N
TTA(I,J)=A(I,J)
Y
TTA(IJ)=1
I=1,121
TS(I)=Ø
TE(I)=Ø
TA(I)=Ø
I=1,20
STT(I) 1
N
Y
J=1,100
TS(I)=TS(I)+TTS(J,I)
I=1,100
10
11

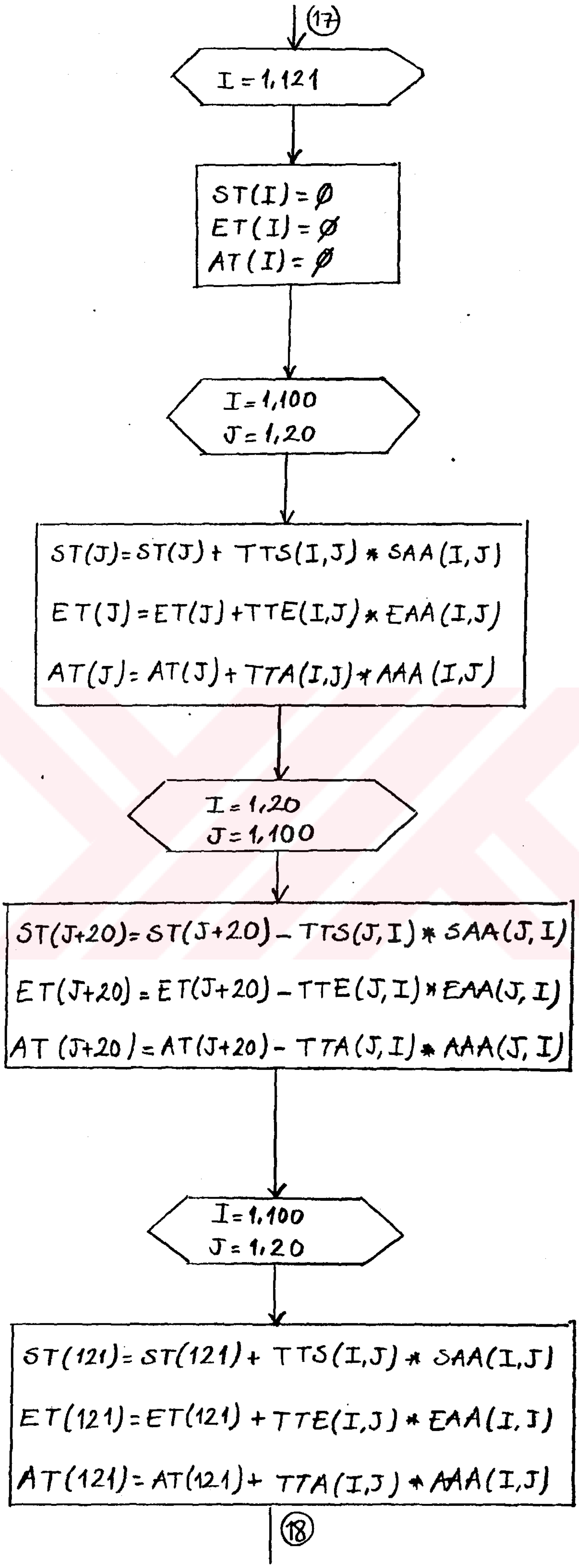


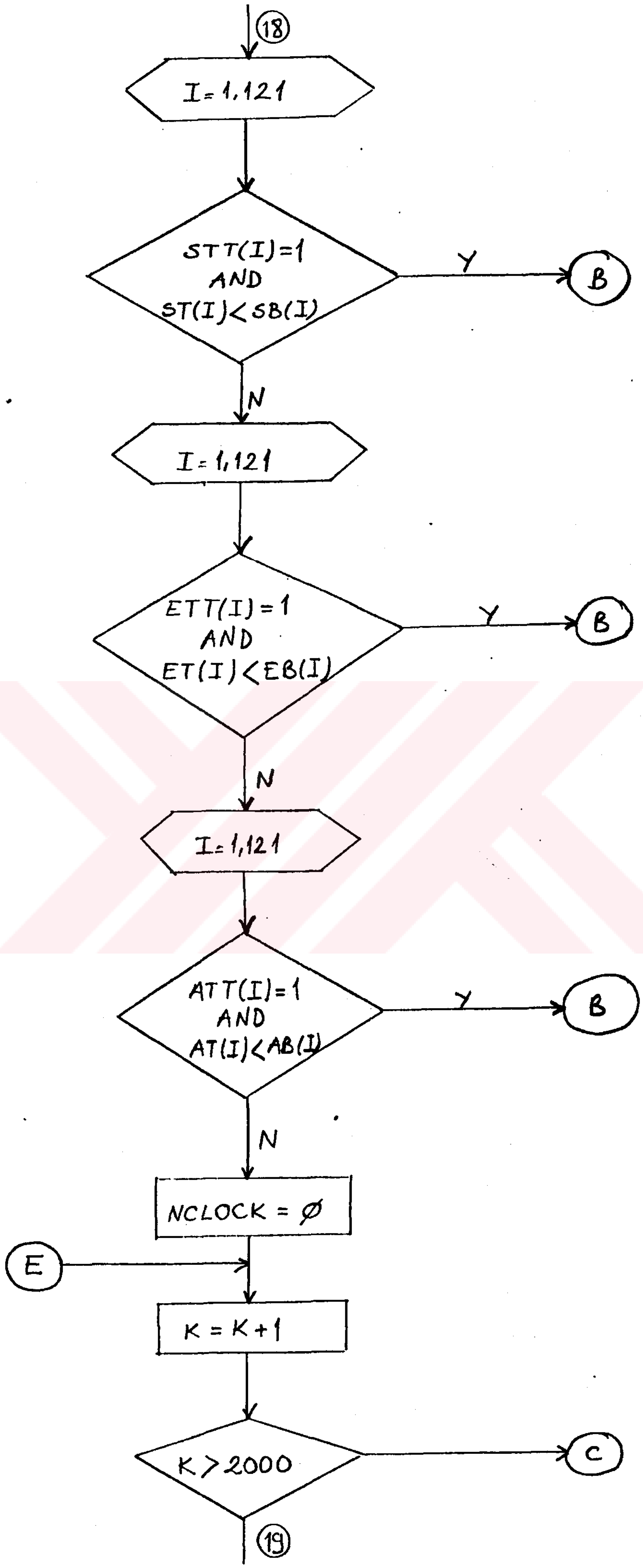
10
11
STT(I+20)=1
N
Y
J=1,20
TS(I+20)=TS(I+20) + TTS(I,J)
STT(121)=1
N
Y
I=1,100
J=1,20
TS(121)=TS(121)+TTS(I,J)
I=1,20
ETT(I)=1
N
Y
J=1,100
12

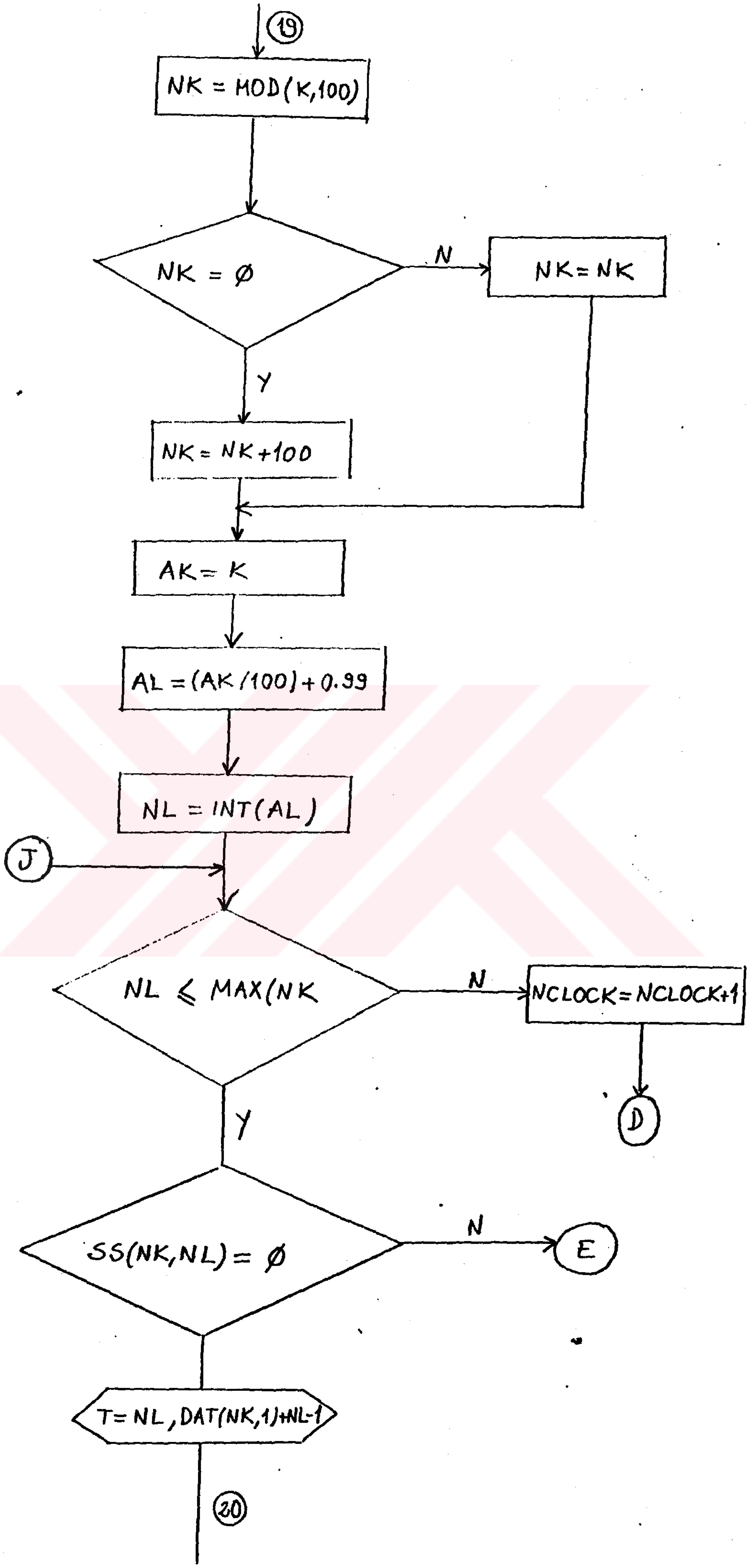

12
TE(I)= TE(I)+ TTE(J,I)
I = 1,100
ETT(I+20)=1
Y
J= 1,20
TE(I+20)= TE(I+20) + TTE(I,J)
ETT(121)=1
N
Y
I = 1,100
J = 1,20
TE(121)= TE(121) + TTE(I,J)
I = 1,20
13
14


13
14
ATT(I)=1
N
Y
J=1,100
TA(I)=TA(I)+TTA(J,I)
I=1,100
ATT(I+20)=1
N
Y
J=1,20
TA(I+20)=TA(I+20)+TTA(I,J)
ATT(121)=1
N
15
16
Y

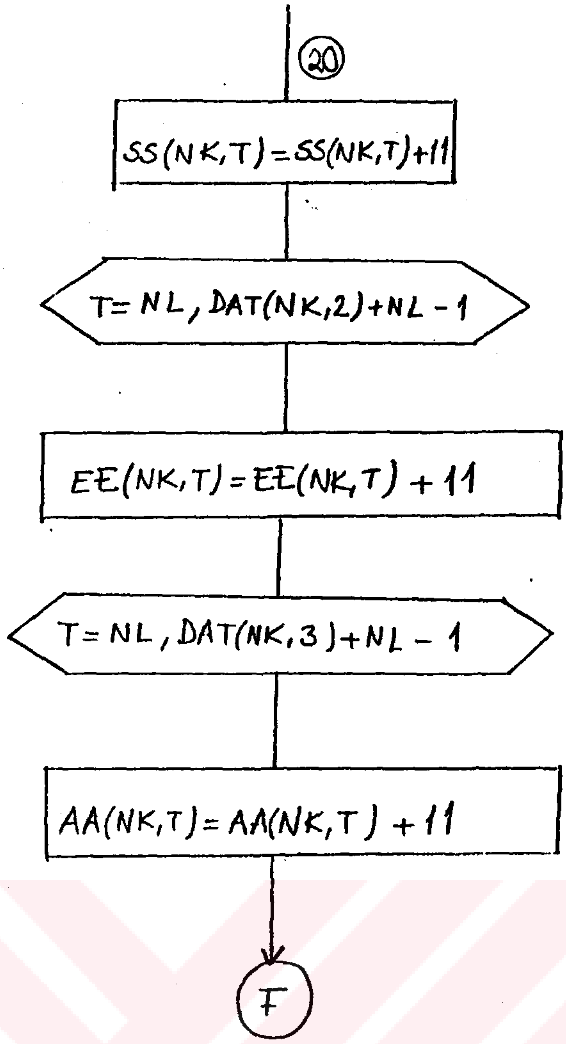

15
16
I=1,100
J=1,20
TA(121)= TA(121)+ TTA(I,J)
I = 1,121
STT(I)=1
AND
TS(I)=∅
Y
B
I=1,121
ETT(I)=1
AND
TE(I)=∅
Y
B
I=1,121
ATT(I)=1
AND
TA(I)=∅
Y
B
17

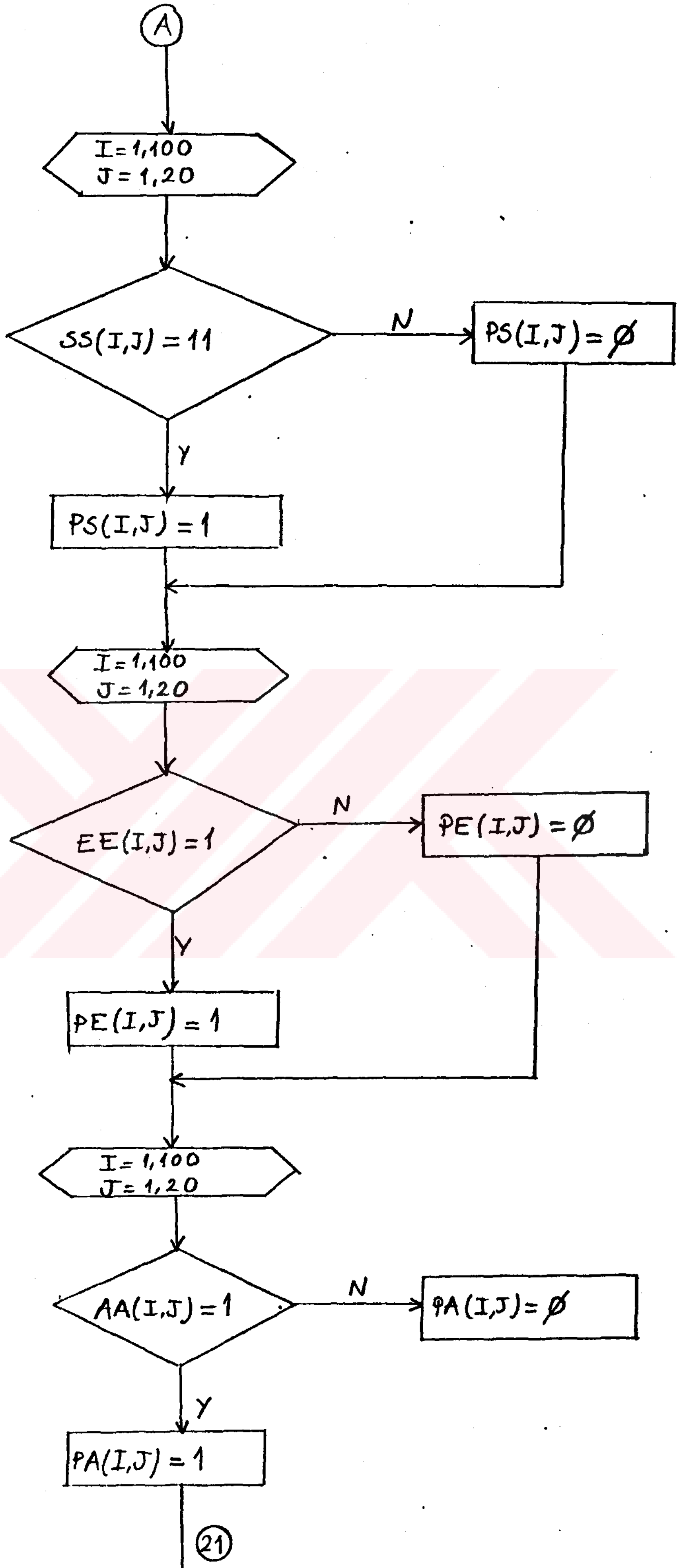

17
I = 1,121
ST(I) = Ø
ET(I) = Ø
AT(I) = Ø
I = 1,100
J = 1,20
ST(J) = ST(J) + TTS(I,J) * SAA(I,J)
ET(J) = ET(J) + TTE(I,J) * EAA(I,J)
AT(J) = AT(J) + TTA(I,J) * AAA(I,J)
I = 1,20
J = 1,100
ST(J+20) = ST(J+20) – TTS(J,I) * SAA(J,I)
ET(J+20) = ET(J+20) – TTE(J,I) * EAA(J,I)
AT(J+20) = AT(J+20) – TTA(J,I) * AAA(J,I)
I = 1,100
J = 1,20
ST(121) = ST(121) + TTS(I,J) * SAA(I,J)
ET(121) = ET(121) + TTE(I,J) * EAA(I,J)
AT(121) = AT(121) + TTA(I,J) * AAA(I,J)
18

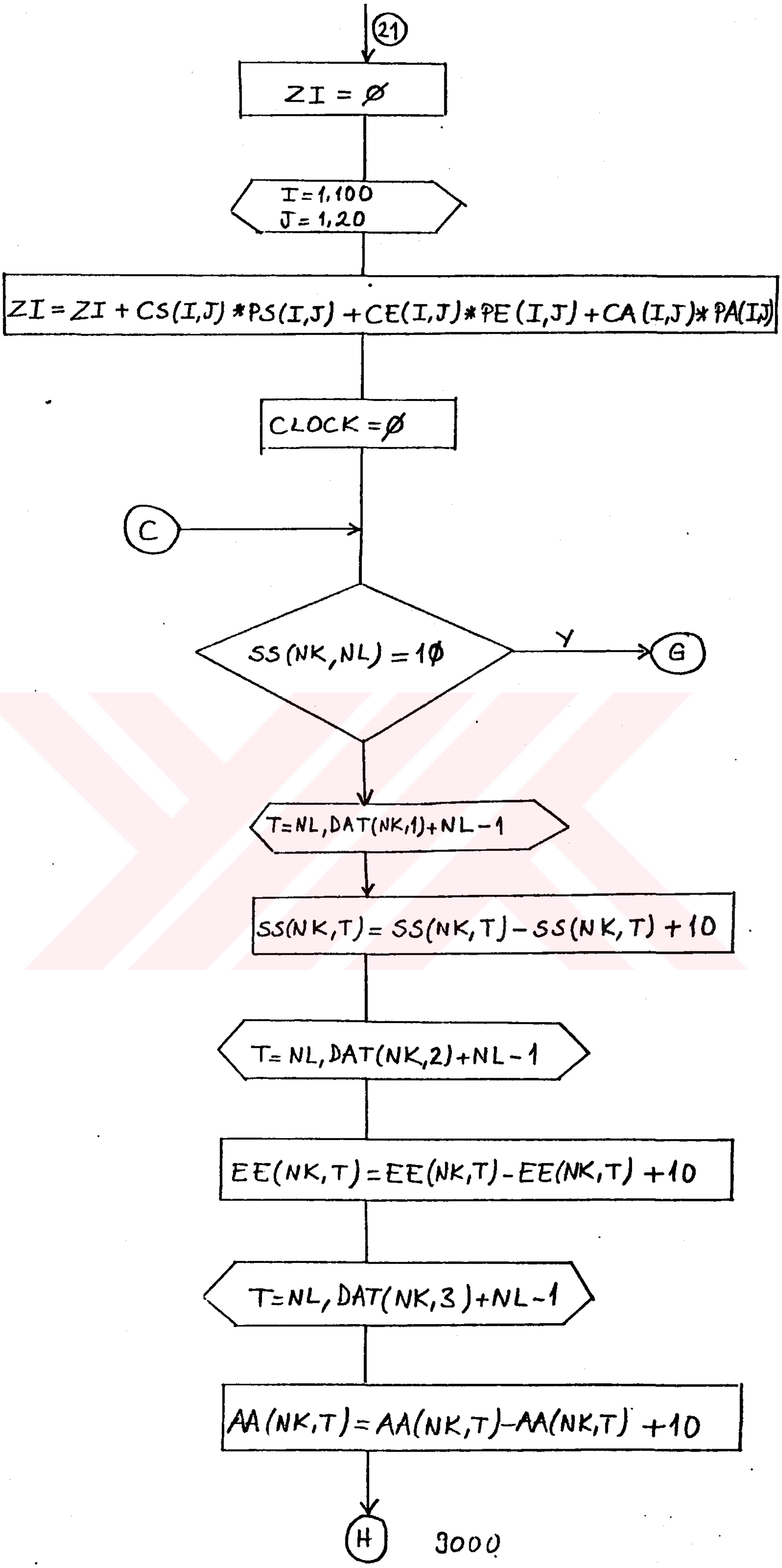

18
I=1,121
STT(I)=1
AND
ST(I)<SB(I)
Y
B
N
I=1,121
ETT(I)=1
AND
ET(I)<EB(I)
Y
B
N
I=1,121
ATT(I)=1
AND
AT(I)<AB(I)
Y
B
N
NCLOCK = Ø
E
K = K+1
K>2000
C
19

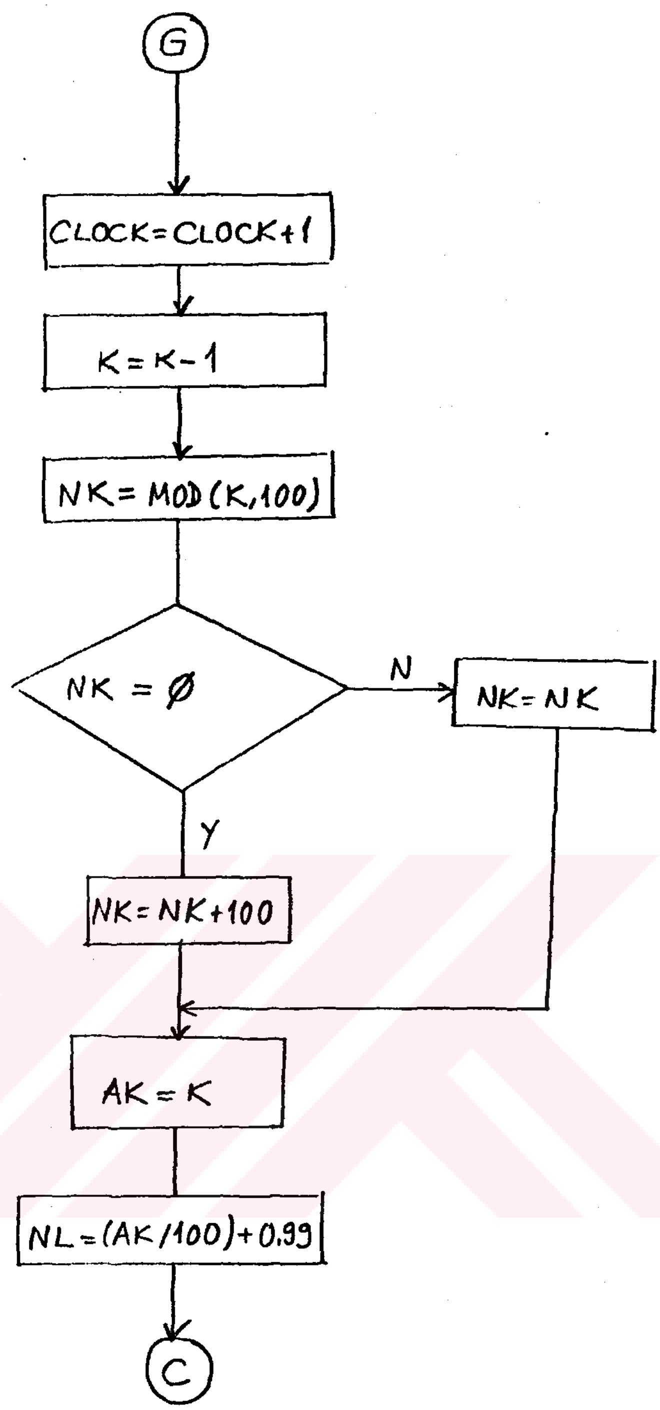


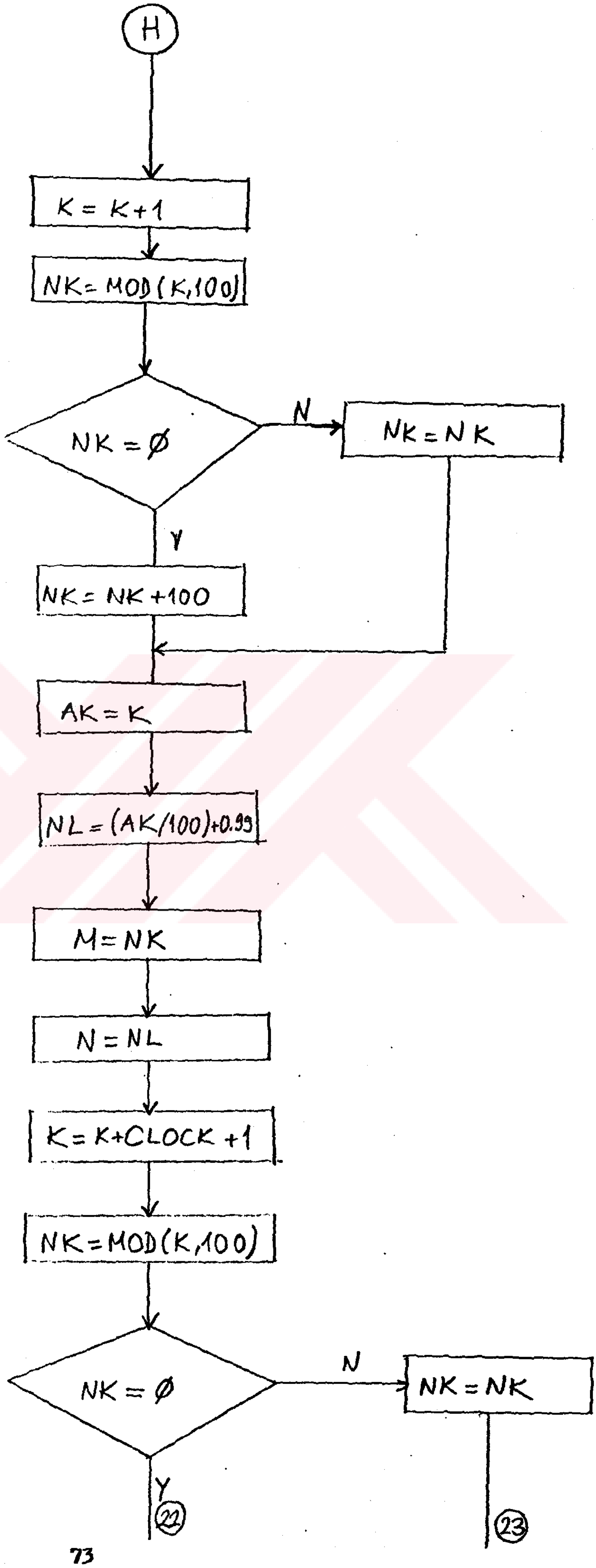
19
NK = MOD(K,100)
NK = Ø
N
NK=NK
Y
NK= NK+100
AK= K
AL = (AK/100)+0.99
NL = INT(AL)
J
NL ≤ MAX(NK
N
NCLOCK=NCLOCK+1
D
Y
SS(NK,NL)= Ø
N
E
T= NL, DAT(NK,1)+NL-1
20

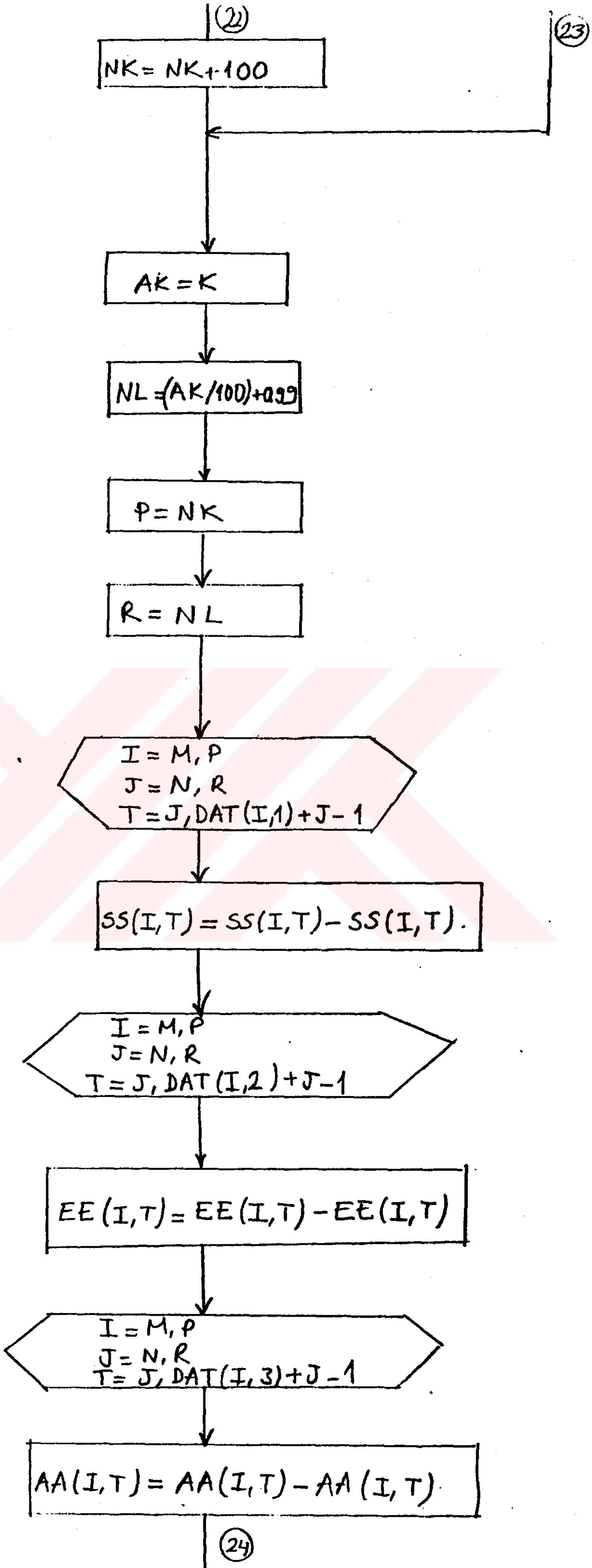

20
SS(NK,T)=SS(NK,T)+11
T=NL, DAT(NK,2)+NL-1
EE(NK,T)=EE(NK,T)+11
T=NL, DAT(NK,3)+NL-1
AA(NK,T)=AA(NK,T)+11
F


A
I = 1,100
J = 1,20
SS(I,J) = 11
N
PS(I,J) = Ø
Y
PS(I,J) = 1
I = 1,100
J = 1,20
EE(I,J) = 1
N
PE(I,J) = Ø
Y
PE(I,J) = 1
I = 1,100
J = 1,20
AA(I,J) = 1
N
PA(I,J) = Ø
Y
PA(I,J) = 1
21

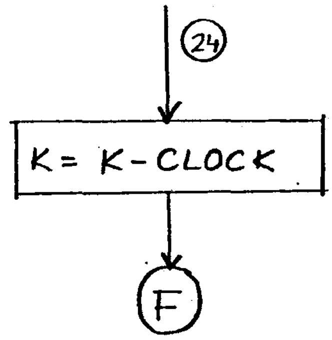

21
ZI = Ø
I = 1,100
J = 1,20
ZI = ZI + CS(I,J) * PS(I,J) + CE(I,J) * PE(I,J) + CA(I,J) * PA(I,J)
CLOCK = Ø
C
SS(NK,NL) = 1Ø
Y
G
T=NL,DAT(NK,1)+NL-1
SS(NK,T) = SS(NK,T) - SS(NK,T) + 10
T= NL,DAT(NK,2)+NL-1
EE(NK,T) = EE(NK,T) - EE(NK,T) + 10
T=NL,DAT(NK,3)+NL-1
AA(NK,T) = AA(NK,T) - AA(NK,T) + 10
H
3000

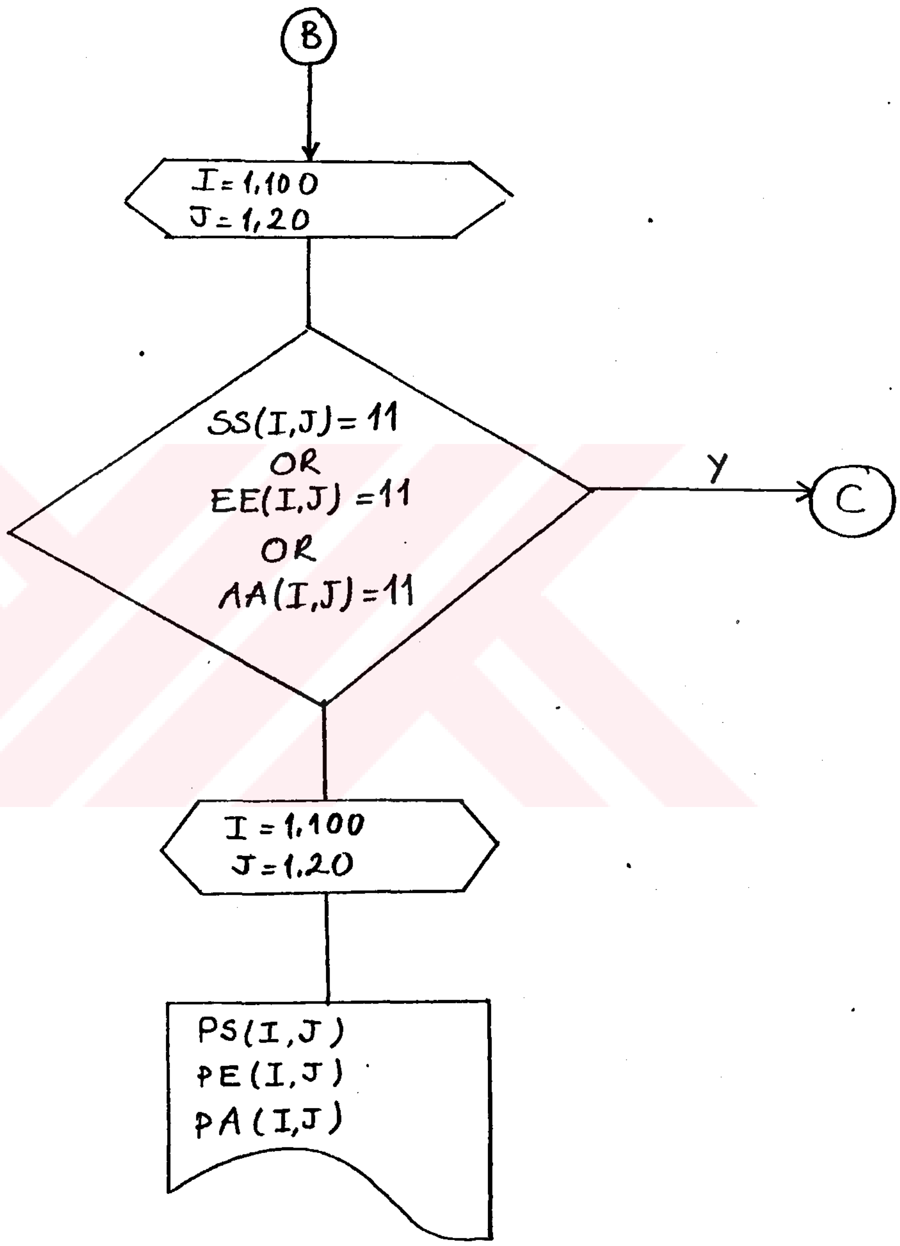

G
CLOCK=CLOCK+1
K=K-1
NK=MOD(K,100)
NK = Ø
N
NK=NK
Y
NK=NK+100
AK=K
NL=(AK/100)+0.99
C

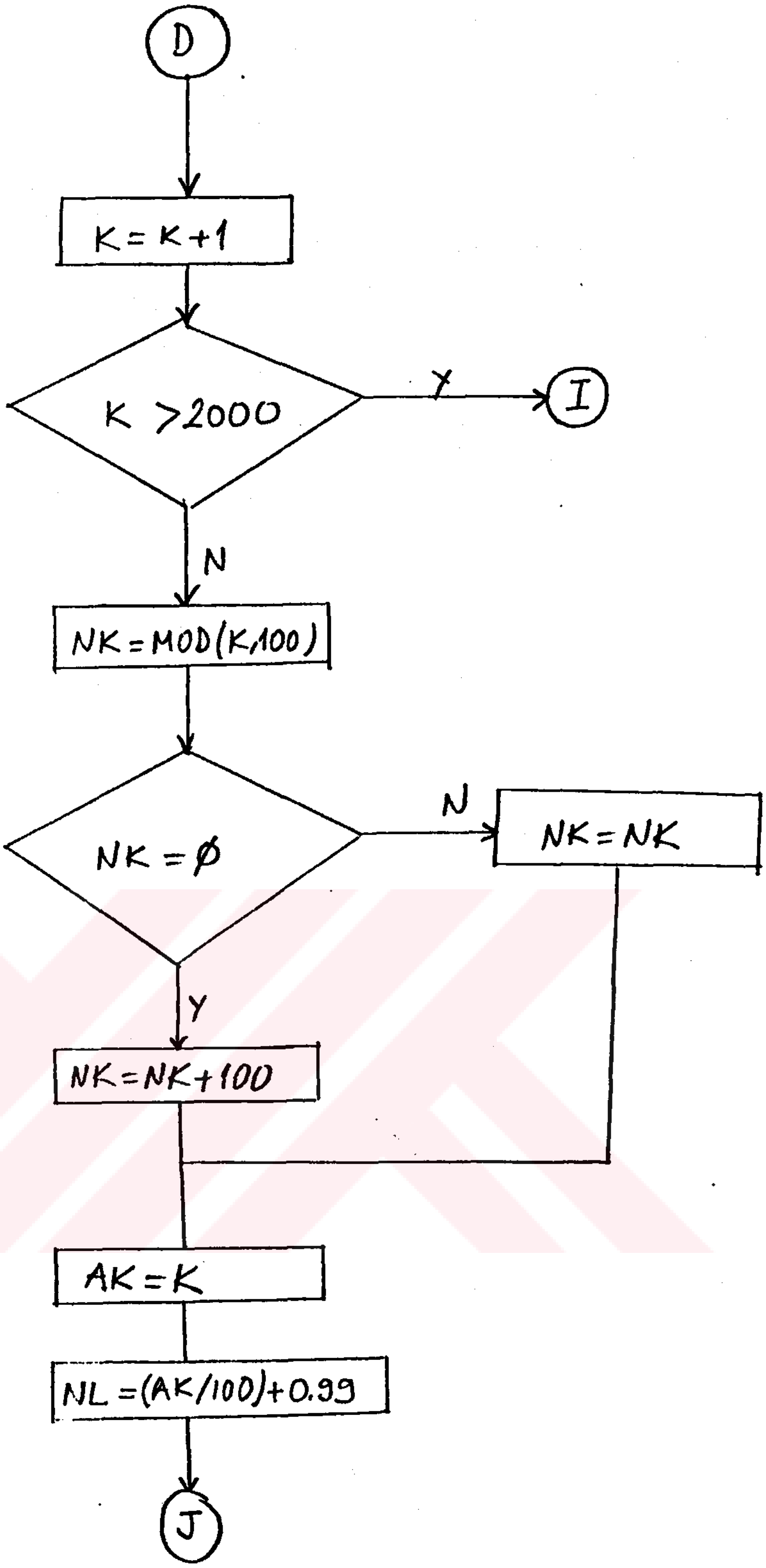

H
K = K+1
NK= MOD (K,100)
NK = Ø
N
NK = NK
Y
NK = NK +100
AK = K
NL = (AK/100)+0.99
M=NK
N = NL
K= K+CLOCK +1
NK=MOD(K,100)
NK = Ø
N
NK = NK
Y
22
23



(22)
(23)
NK = NK + 100
AK = K
NL = (AK/100) + 0.99
P = NK
R = NL
I = M, P
J = N, R
T = J, DAT (I,1) + J - 1
SS(I,T) = SS(I,T) - SS(I,T).
I = M, P
J = N, R
T = J, DAT (I,2) + J - 1
EE (I,T) = EE (I,T) - EE (I,T)
I = M, P
J = N, R
T = J, DAT (I,3) + J - 1
AA (I,T) = AA (I,T) - AA (I,T)
(24)


24
K= K-CLOCK
F


B
I=1,100
J=1,20
SS(I,J)=11
OR
EE(I,J)=11
OR
AA(I,J)=11
Y
C
I=1,100
J=1,20
PS(I,J)
PE(I,J)
PA(I,J)


D
K=K+1
K >2000
Y
I
N
NK=MOD(K,100)
NK=Ø
N
NK=NK
Y
NK=NK+100
AK=K
NL=(AK/100)+0.99
J

```
DO=-LONG/-OT,ARG=-COMMON/-FIXED,CS= USER/-FIXED,DB=-TB/-SB/-SL/ ER/-ID/-PMD/-ST,-AL,PL=5000
FTN5,I=T1,L=T03.

C     SOLVING 0-1 LP PROBLEM VIA IMPLICIT ENUMERATION ALGORITHM.
      PROGRAM TEZ1(TD2,T03,TAPE7=TD2,TAPE6=T03)
      INTEGER S(100,20),E(100,20),A(100,20),DAT(100,3)
     &,SAA(100,20),EAA(100,20),AAA(100,20),CS(100,20),CE(100,20)
     &,CA(100,20),MAX(100),SS(100,20),EE(100,20),AA(100,20)
     &,VS(100,20),VE(100,20),VA(100,20),SB(121),EB(121),AB(121)
     &,SD(121),ED(121),AD(121),SDD(121),EDD(121),ADD(121),STT(121)
     &,ETT(121),ATT(121),TTS(100,20),TTE(100,20),TTA(100,20),ST(121)
     &,ET(121),AT(121),TS(121),TE(121),TA(121),PS(100,20),PE(100,20)
     &,PA(100,20),NK,NL
      READ(7,1) ((DAT(I,J),J=1,3),I=1,100)
      READ(7,4) (SB(I),I=1,121)
      READ(7,4) (EB(I),I=1,121)
      READ(7,4) (AB(I),I=1,121)
      READ(7,6) ((SAA(I,J),J=1,20),I=1,100)
      READ(7,6) ((EAA(I,J),J=1,20),I=1,100)
      READ(7,6) ((AAA(I,J),J=1,20),I=1,100)
      READ(7,6) ((CS(I,J),J=1,20),I=1,100)
      READ(7,6) ((CE(I,J),J=1,20),I=1,100)
      READ(7,6) ((CA(I,J),J=1,20),I=1,100)
      READ(7,2) (MAX(I),I=1,100)
    1 FORMAT (I1,1X,I1,1X,I1)
    2 FORMAT (I2)
    4 FORMAT (I5)
    6 FORMAT ((15X,19(I1,1X),I1))
    7 FORMAT (I6)
      II=6000
      K=0
      DO 10 J=1,20
      DO 20 I=1,100
      SS(I,J)=0
      EE(I,J)=0
      AA(I,J)=0
   20 CONTINUE
   10 CONTINUE
    5 DO 30 J=1,20
      DO 40 I=1,100
      IF(SS(I,J)-11) 31,32,32
   32 S(I,J)=1
      GO TO 40
   31 S(I,J)=0
   40 CONTINUE
   30 CONTINUE
      DO 50 J=1,20
      DO 60 I=1,100
      IF(EE(I,J)-11) 51,52,52
   52 E(I,J)=1
      GO TO 60
   51 E(I,J)=0
   60 CONTINUE
   50 CONTINUE
      DO 70 J=1,20
      DO 80 I=1,100
      IF(AA(I,J)-11) 71,72,72
   72 A(I,J)=1
```

```
      GO TO 80
   71 A(I,J)=0
   80 CONTINUE
   70 CONTINUE
C     FIND V,THE SET OF CONSTRAINTS VIOLATED WHEN PARTIAL SOLUTION
C     S IS COMPLETED BY SETTING TO ZERO ALL VARIABLES NOT IN S.
      DO 90 J=1,20
      DO 100 I=1,100
      IF(SS(I,J)-0) 91,92,91
   92 VS(I,J)=0
      GO TO 100
   91 VS(I,J)=S(I,J)
  100 CONTINUE
   90 CONTINUE
      DO 110 J=1,20
      DO 120 I=1,100
      IF(EE(I,J)-0) 111,112,111
  112 VE(I,J)=0
      GO TO 120
  111 VE(I,J)=E(I,J)
  120 CONTINUE
  110 CONTINUE
      DO 130 J=1,20
      DO 140 I=1,100
      IF(AA(I,J)-0) 131,132,131
  132 VA(I,J)=0
      GO TO 140
  131 VA(I,J)=A(I,J)
  140 CONTINUE
  130 CONTINUE
      DO 150 I=1,121
      SD(I)=0
      ED(I)=0
      AD(I)=0
  150 CONTINUE
      DO 160 J=1,20
      DO 170 I=1,100
      SD(J)=SD(J)+VS(I,J)*SAA(I,J)
  170 CONTINUE
  160 CONTINUE
      DO 161 J=1,100
      DO 162 I=1,20
      SD(J+20)=SD(J+20)-VS(J,I)*SAA(J,I)
  162 CONTINUE
  161 CONTINUE
      DO 163 I=1,100
      DO 164 J=1,20
      SD(121)=SD(121)+VS(I,J)*SAA(I,J)
  164 CONTINUE
  163 CONTINUE
      DO 180 J=1,20
      DO 190 I=1,100
      ED(J)=ED(J)+VE(I,J)*EAA(I,J)
  190 CONTINUE
  180 CONTINUE
      DO 181 J=1,100
      DO 182 I=1,20
```

```
      ED(J+20)=ED(J+20)-VE(J,I)*EAA(J,I)
  182 CONTINUE
  181 CONTINUE
      DO 183 I=1,100
      DO 184 J=1,20
      ED(121)=ED(121)+VE(I,J)*EAA(I,J)
  184 CONTINUE
  183 CONTINUE
      DO 200 J=1,20
      DO 210 I=1,100
      AD(J)=AD(J)+VA(I,J)*AAA(I,J)
  210 CONTINUE
  200 CONTINUE
      DO 201 J=1,100
      DO 202 I=1,20
      AD(J+20)=AD(J+20)-VA(J,I)*AAA(J,I)
  202 CONTINUE
  201 CONTINUE
      DO 203 I=1,100
      DO 204 J=1,20
      AD(121)=AD(121)+VA(I,J)*AAA(I,J)
  204 CONTINUE
  203 CONTINUE
      DO 220 I=1,121
      IF(SD(I).GE.SB(I)) THEN
      SDD(I)=0
      ELSE
      SDD(I)=1
      ENDIF
  220 CONTINUE
      DO 230 I=1,121
      IF(ED(I).GE.EB(I)) THEN
      EDD(I)=0
      ELSE
      EDD(I)=1
      ENDIF
  230 CONTINUE
      DO 240 I=1,121
      IF(AD(I).GE.AB(I)) THEN
      ADD(I)=0
      ELSE
      ADD(I)=1
      ENDIF
  240 CONTINUE
      TDD=0
      DO 250 I=1,121
      TDD=TDD+SDD(I)+EDD(I)+ADD(I)
  250 CONTINUE
C     IS V EMPTY ?
      IF(TDD.EQ.0) GO TO 7500
C     SET THE OBJECTIVE FUNCTION COEFFICIENT LIMIT TO ZBAP-FP
      FP=0
      DO 260 J=1,20
      DO 270 I=1,100
      FP=FP+CS(I,J)*VS(I,J)+CE(I,J)*VE(I,J)+CA(I,J)*VA(I,J)
  270 CONTINUE
  260 CONTINUE
```

```
      Z=ZI-FP
C     STORE IN THE SET T(I) FOR I INCLUDED IN V EACH VARIABLE NOT
C     IN S WHICH HAS:
C     1-)AN OBJECTIVE FUNCTION COEFFICIENT LESS THAN THE LIMIT.
C     2-)A POSITIVE COEFFICIENT IN CONSTRAINT I IN V.
      DO 280 I=1,121
      IF(SDD(I).EQ.1) THEN
      STT(I)=1
      ELSE
      STT(I)=0
      ENDIF
  280 CONTINUE
      DO 290 I=1,121
      IF(EDD(I).EQ.1) THEN
      ETT(I)=1
      ELSE
      ETT(I)=0
      ENDIF
  290 CONTINUE
      DO 300 I=1,121
      IF(ADD(I).EQ.1) THEN
      ATT(I)=1
      ELSE
      ATT(I)=0
      ENDIF
  300 CONTINUE
      DO 310 J=1,20
      DO 320 I=1,100
      IF(STT(J).EQ.1.AND.SS(I,J).EQ.0) THEN
      TTS(I,J)=1
      ELSE
      TTS(I,J)=S(I,J)
      ENDIF
  320 CONTINUE
  310 CONTINUE
      DO 330 J=1,20
      DO 340 I=1,100
      IF(ETT(J).EQ.1.AND.EE(I,J).EQ.0) THEN
      TTE(I,J)=1
      ELSE
      TTE(I,J)=E(I,J)
      ENDIF
  340 CONTINUE
  330 CONTINUE
      DO 350 J=1,20
      DO 360 I=1,100
      IF(ATT(J).EQ.1.AND.AA(I,J).EQ.0) THEN
      TTA(I,J)=1
      ELSE
      TTA(I,J)=A(I,J)
      ENDIF
  360 CONTINUE
  350 CONTINUE
      DO 352 J=1,20
      DO 353 I=1,100
      IF(STT(121).EQ.1.AND.SS(I,J).EQ.0) THEN
      TTS(I,J)=1
```

```
      ELSE
      TTS(I,J)=S(I,J)
      ENDIF
  353 CONTINUE
  352 CONTINUE
      DO 354 J=1,20
      DO 355 I=1,100
      IF(ETT(121).EQ.1.AND.EE(I,J).EQ.0) THEN
      TTE(I,J)=1
      ELSE
      TTE(I,J)=E(I,J)
      ENDIF
  355 CONTINUE
  354 CONTINUE
      DO 356 J=1,20
      DO 357 I=1,100
      IF(ATT(121).EQ.1.AND.AA(I,J).EQ.0) THEN
      TTA(I,J)=1
      ELSE
      TTA(I,J)=A(I,J)
      ENDIF
  357 CONTINUE
  356 CONTINUE
      DO 351 I=1,121
      TS(I)=0
      TE(I)=0
      TA(I)=0
  351 CONTINUE
      DO 370 I=1,20
      IF(STT(I).EQ.1) GO TO 1000
      GO TO 370
 1000 DO 375 J=1,100
      TS(I)=TS(I)+TTS(J,I)
  375 CONTINUE
  370 CONTINUE
      DO 371 I=1,100
      IF(STT(I+20).EQ.1) GO TO 1005
      GO TO 371
 1005 DO 376 J=1,20
      TS(I+20)=TS(I+20)+TTS(I,J)
  376 CONTINUE
  371 CONTINUE
      IF(STT(121).EQ.1) GO TO 1010
      GO TO 1020
 1010 DO 372 I=1,100
      DO 373 J=1,20
      TS(121)=TS(121)+TTS(I,J)
  373 CONTINUE
  372 CONTINUE
 1020 DO 380 I=1,20
      IF(ETT(I).EQ.1) GO TO 1500
      GO TO 380
 1500 DO 385 J=1,100
      TE(I)=TE(I)+TTE(J,I)
  385 CONTINUE
  380 CONTINUE
      DO 381 I=1,100
```

```
      IF(ETT(I+20).EQ.1) GO TO 1505
      GO TO 381
 1505 DO 386 J=1,20
      TE(I+20)=TE(I+20)+TTE(I,J)
  386 CONTINUE
  381 CONTINUE
      IF(ETT(121).EQ.1) GO TO 1510
      GO TO 1520
 1510 DO 382 I=1,100
      DO 383 J=1,20
      TE(121)=TE(121)+TTE(I,J)
  383 CONTINUE
  382 CONTINUE
 1520 DO 390 I=1,20
      IF(ATT(I).EQ.1) GO TO 2000
      GO TO 390
 2000 DO 395 J=1,100
      TA(I)=TA(I)+TTA(J,I)
  395 CONTINUE
  390 CONTINUE
      DO 391 I=1,100
      IF(ATT(I+20).EQ.1) GO TO 2005
      GO TO 391
 2005 DO 396 J=1,20
      TA(I+20)=TA(I+20)+TTA(I,J)
  396 CONTINUE
  391 CONTINUE
      IF(ATT(121).EQ.1) GO TO 2010
      GO TO 2020
 2010 DO 392 I=1,100
      DO 393 J=1,20
      TA(121)=TA(121)+TTA(I,J)
  393 CONTINUE
  392 CONTINUE
C     ARE SOME T(I) EMPTY ?
 2020 DO 400 I=1,121
      IF(STT(I).EQ.1.AND.TS(I).EQ.0) GO TO 5000
  400 CONTINUE
      DO 410 I=1,121
      IF(ETT(I).EQ.1.AND.TE(I).EQ.0) GO TO 5000
  410 CONTINUE
      DO 420 I=1,121
      IF(ATT(I).EQ.1.AND.TA(I).EQ.0) GO TO 5000
  420 CONTINUE
      DO 430 I=1,121
      ST(I)=0
      ET(I)=0
      AT(I)=0
  430 CONTINUE
      DO 440 J=1,20
      DO 450 I=1,100
      ST(J)=ST(J)+TTS(I,J)*SAA(I,J)
      ET(J)=ET(J)+TTE(I,J)*EAA(I,J)
      AT(J)=AT(J)+TTA(I,J)*AAA(I,J)
  450 CONTINUE
  440 CONTINUE
      DO 441 J=1,100
```

```
      DO 442 I=1,20
      ST(J+20)=ST(J+20)-TTS(J,I)*SAA(J,I)
      ET(J+20)=ET(J+20)-TTE(J,I)*EAA(J,I)
      AT(J+20)=AT(J+20)-TTA(J,I)*AAA(J,I)
  442 CONTINUE
  441 CONTINUE
      DO 443 I=1,100
      DO 444 J=1,20
      ST(121)=ST(121)+TTS(I,J)*SAA(I,J)
      ET(121)=ET(121)+TTE(I,J)*EAA(I,J)
      AT(121)=AT(121)+TTA(I,J)*AAA(I,J)
  444 CONTINUE
  443 CONTINUE
C     CAN EVERY CONSTRAINT I IN V BE MADE FEASIBLE BY ADDING ONLY
C     VARIABLES IN T(I) ?
      DO 460 I=1,121
      IF(STT(I).EQ.1.AND.ST(I).LT.SB(I)) GO TO 5000
  460 CONTINUE
      DO 470 I=1,121
      IF(ETT(I).EQ.1.AND.ET(I).LT.EB(I)) GO TO 5000
  470 CONTINUE
      DO 480 I=1,121
      IF(ATT(I).EQ.1.AND.AT(I).LT.AB(I)) GO TO 5000
  480 CONTINUE
      NCLOCK=0
 3000 K=K+1
      IF(K.GT.2000) GO TO 8000
      NK=MOD(K,100)
      IF(NK.EQ.0) THEN
      NK=NK+100
      ELSE
      NK=NK
      ENDIF
      AK=K
      AL=(AK/100)+0.99
      NL=INT(AL)
 4650 IF(NL.LE.MAX(NK)) GO TO 4500
      NCLOCK=NCLOCK+1
      GO TO 7650
C     ADD TO S THE VARIABLE IN T(I) WITH THE GREATEST COEFFICIENT SUM.
 4500 IF(SS(NK,NL).NE.0) GO TO 3000
      DO 490 T=NL,DAT(NK,1)+NL-1
      SS(NK,T)=SS(NK,T)-SS(NK,T)+11
  490 CONTINUE
      DO 500 T=NL,DAT(NK,2)+NL-1
      EE(NK,T)=EE(NK,T)-EE(NK,T)+11
  500 CONTINUE
      DO 510 T=NL,DAT(NK,3)+NL-1
      AA(NK,T)=AA(NK,T)-AA(NK,T)+11
  510 CONTINUE
      GO TO 5
C     COMPLETE THE PARTIAL SOLUTION S BY SETTING TO ZERO ALL
C     VARIABLES NOT IN S.THIS COMPLETED SOLUTION BECOMES THE
C     INCUMBENT SOLUTION XBAR AND THE VALUE OF THE OBJECTIVE
C     FUNCTION AT XBAR BECOMES THE NEW VALUE OF ZBAR.
 7500 DO 520 J=1,20
      DO 530 I=1,100
```

```
      IF(SS(I,J).EQ.11) THEN
      PS(I,J)=1
      ELSE
      PS(I,J)=0
      ENDIF
530   CONTINUE
520   CONTINUE
      DO 540 J=1,20
      DO 550 I=1,100
      IF(EE(I,J).EQ.11) THEN
      PE(I,J)=1
      ELSE
      PE(I,J)=0
      ENDIF
550   CONTINUE
540   CONTINUE
      DO 560 J=1,20
      DO 570 I=1,100
      IF(AA(I,J).EQ.11) THEN
      PA(I,J)=1
      ELSE
      PA(I,J)=0
      ENDIF
570   CONTINUE
560   CONTINUE
      ZI=0
      DO 580 J=1,20
      DO 590 I=1,100
      ZI=ZI+CS(I,J)*PS(I,J)+CE(I,J)*PE(I,J)+CA(I,J)*PA(I,J)
590   CONTINUE
580   CONTINUE
      CLOCK=0
C     LOCATE THE RIGHTMOST ELEMENT IN S. REPLACE IT WITH ITS
C     (NEGATIVE) COMPLEMENTS AND DROP ANY ELEMENTS TO THE RIGHT.
8000  IF(SS(NK,NL).EQ.10) GO TO 8500
      DO 600 T=NL,DAT(NK,1)+NL-1
      SS(NK,T)=SS(NK,T)-SS(NK,T)+10
600   CONTINUE
      DO 610 T=NL,DAT(NK,2)+NL-1
      EE(NK,T)=EE(NK,T)-EE(NK,T)+10
610   CONTINUE
      DO 620 T=NL,DAT(NK,3)+NL-1
      AA(NK,T)=AA(NK,T)-AA(NK,T)+10
620   CONTINUE
      GO TO 9000
8500  CLOCK=CLOCK+1
      K=K-1
      NK=MOD(K,100)
      IF(NK.EQ.0) THEN
      NK=NK+100
      ELSE
      NK=NK
      ENDIF
      AK=K
      NL=(AK/100)+0.99
      GO TO 8000
9000  K=K+1
```

```
      NK=MOD(K,100)
      IF(NK.EQ.0) THEN
      NK=NK+100
      ELSE
      NK=NK
      ENDIF
      AK=K
      NL=(AK/100)+0.99
      M=NK
      N=NL
      K=K+CLOCK-1
      NK=MOD(K,100)
      IF(NK.EQ.0) THEN
      NK=NK+100
      ELSE
      NK=NK
      ENDIF
      AK=K
      NL=(AK/100)+0.99
      P=NK
      R=NL
      DO 630 I=M,P
      DO 640 J=N,R
      DO 641 T=J,DAT(I,1)+J-1
      SS(I,T)=SS(I,T)-SS(I,T)
  641 CONTINUE
  640 CONTINUE
  630 CONTINUE
      DO 700 I=M,P
      DO 710 J=N,R
      DO 711 T=J,DAT(I,2)+J-1
      EE(I,T)=EE(I,T)-EE(I,T)
  711 CONTINUE
  710 CONTINUE
  700 CONTINUE
      DO 720 I=M,P
      DO 730 J=N,R
      DO 721 T=J,DAT(I,3)+J-1
      AA(I,T)=AA(I,T)-AA(I,T)
  721 CONTINUE
  730 CONTINUE
  720 CONTINUE
      K=K-CLOCK
      GO TO 5
C     ARE ALL ELEMENTS IN S NEGATIVE ?
 5000 DO 650 K=1,20
      DO 660 L=1,100
      IF(SS(L,K).EQ.11.OR.EE(L,K).EQ.11.OR.AA(L,K).EQ.11) GO TO 8000
C     TERMINATE.THE INCUMBENT SOLUTION SOLUTON IF ANY,IS AN OPTIMAL
C     SOLUTION.IF NO INCUMBENT SOLUTION HAS BEEN FOUND,THERE IS NO
C     FEASIBLE SOLUTION BETTER THAN.THE SOLUTION CORRESPONDING TO
C     THE BEST KNOWN UPPER BOUND.
  660 CONTINUE
  650 CONTINUE
      DO 651 J=1,20
      DO 652 I=1,100
      PS(I,J)=1-PS(I,J)
```

```
            PE(I,J)=1-PE(I,J)
            PA(I,J)=1-PA(I,J)
  652       CONTINUE
  651       CONTINUE
            WRITE(6,8)
            WRITE(6,6) ((PS(I,J),J=1,20),I=1,100)
            WRITE(6,9)
            WRITE(6,6) ((PE(I,J),J=1,20),I=1,100)
            WRITE(6,11)
            WRITE(6,6) ((PA(I,J),J=1,20),I=1,100)
            GO TO 6500
    3       FORMAT (5X,"SENIOR JOBS",12X,"WEEKS")
    9       FORMAT (4X,"EXP.ASS.JOBS",12X,"WEEKS")
   11       FORMAT (8X,"ASS.JOBS",12X,"WEEKS")
 9500       K=K-NCLOCK
            NK=MOD(K,100)
            IF(NK.EQ.0) THEN
            NK=NK+100
            ELSE
            NK=NK
            ENDIF
            AK=K
            NL=(AK/100)+0.99
            GO TO 8000
 7650       K=K+1
            IF(K.GT.2000) GO TO 9500
            NK=MOD(K,100)
            IF(NK.EQ.0) THEN
            NK=NK+100
            ELSE
            NK=NK
            ENDIF
            AK=K
            NL=(AK/100)+0.99
            GO TO 4650
 6500       STOP
            END
```

--VARIABLE MAP--(LO=A)

NAME	ADDRESS	BLOCK	PROPERTIES	TYPE	SIZE
A	14316B			INTEGER	2000
AA	30256B			INTEGER	2000
AAA	30352B			INTEGER	2000
AB	100340B			INTEGER	121
AD	101113B			INTEGER	121
ADD	101566B			INTEGER	121
AK	[illegible]			REAL	
AL	133677B			REAL	
AT	116774B			INTEGER	121
ATT	102441B			INTEGER	121
CA	44372B			INTEGER	2000
CE	40412B			INTEGER	2000
CLOCK	133714B			REAL	
CS	34472B			INTEGER	2000
DAT	20236B			INTEGER	300

NAME	ADDRESS	BLOCK	PROPERTIES	TYPE	SIZE
E	10376B			INTEGER	2000
EAA	24632B			INTEGER	2000
EB	100147B			INTEGER	121
ED	100722B			INTEGER	121
EDD	101475B			INTEGER	121
EE	54336B			INTEGER	2000
ET	116603B			INTEGER	121
ETT	102250B			INTEGER	121
FP	133612B			REAL	
I	133522B			INTEGER	
J	133523B			INTEGER	
K	133543B			INTEGER	
L	133736B			INTEGER	
M	133720B			INTEGER	
MAX	50252B			INTEGER	100

NAME	ADDRESS	BLOCK	PROPERTIES	TYPE	SIZE
N	133721B			INTEGER	
NCLOCK	133675B			INTEGER	
NK	133520B			INTEGER	
NL	133521B			INTEGER	
P	133722B			REAL	
PA	127600B			INTEGER	2000
PE	123660B			INTEGER	2000
PS	117740B			INTEGER	2000
R	133723B			REAL	
S	4456B			INTEGER	2000
SAA	20712B			INTEGER	2000
SB	77756B			INTEGER	121
SD	100531B			INTEGER	121
SDD	101304B			INTEGER	121
SS	50416B			INTEGER	2000
ST	116412B			INTEGER	121
STT	102057B			INTEGER	121
T	133700B			REAL	
TA	117547B			INTEGER	121
TDD	133610B			REAL	
TE	117356B			INTEGER	121
TS	117165B			INTEGER	121
TTA	112472B			INTEGER	2000
TTE	106552B			INTEGER	2000
TTS	102632B			INTEGER	2000
VA	74036B			INTEGER	2000
VE	70116B			INTEGER	2000
VS	64176B			INTEGER	2000
Z	133615B		*S*	REAL	
ZI	133542B			REAL	

--PROCEDURES--(LO=A)

NAME	TYPE	ARGS	CLASS
INT	GENERIC	1	INTRINSIC
MOD	GENERIC	2	INTRINSIC

87

--STATEMENT LABELS--(LO=A)

LABEL	ADDRESS	PROPERTIES	DEF
1	4275B	FORMAT	22
2	4301B	FORMAT	23
4	4303B	FORMAT	24
5	373B		36
6	4305B	FORMAT	25
7	4310B	FORMAT	26
8	4312B	FORMAT	523
9	4317B	FORMAT	524
10	INACTIVE	DO-TERM	35
11	4324B	FORMAT	525
20	INACTIVE	DO-TERM	34
30	INACTIVE	DO-TERM	43
31	420B		41
32	INACTIVE		39
40	474B	DO-TERM	42
50	INACTIVE	DO-TERM	51
51	461B		47
52	INACTIVE		47
60	465B	DO-TERM	50
70	INACTIVE	DO-TERM	59
71	522B		57
72	INACTIVE		55
80	520B	DO-TERM	53
90	INACTIVE	DO-TERM	69
91	562B		67
92	INACTIVE		65
100	567B	DO-TERM	63
110	INACTIVE	DO-TERM	77
111	623B		75
112	INACTIVE		73
120	630B	DO-TERM	76
130	INACTIVE	DO-TERM	85
131	664B		83
132	INACTIVE		81
140	671B	DO-TERM	84
150	INACTIVE	DO-TERM	90
160	INACTIVE	DO-TERM	95
161	INACTIVE	DO-TERM	100
162	INACTIVE	DO-TERM	99
163	INACTIVE	DO-TERM	105
164	INACTIVE	DO-TERM	104
170	INACTIVE	DO-TERM	94
180	INACTIVE	DO-TERM	110
181	INACTIVE	DO-TERM	115
182	INACTIVE	DO-TERM	114
183	INACTIVE	DO-TERM	120
184	INACTIVE	DO-TERM	119
190	INACTIVE	DO-TERM	109
200	INACTIVE	DO-TERM	125
201	INACTIVE	DO-TERM	130
202	INACTIVE	DO-TERM	129
203	INACTIVE	DO-TERM	135
204	INACTIVE	DO-TERM	134
210	INACTIVE	DO-TERM	124
220	INACTIVE	DO-TERM	142
230	INACTIVE	DO-TERM	149
240	INACTIVE	DO-TERM	156
250	INACTIVE	DO-TERM	160
260	INACTIVE	DO-TERM	169
270	INACTIVE	DO-TERM	168
280	INACTIVE	DO-TERM	181
290	INACTIVE	DO-TERM	188
300	INACTIVE	DO-TERM	195
310	INACTIVE	DO-TERM	204
320	INACTIVE	DO-TERM	203
330	INACTIVE	DO-TERM	213
340	INACTIVE	DO-TERM	212
350	INACTIVE	DO-TERM	222
351	INACTIVE	DO-TERM	254
352	INACTIVE	DO-TERM	231
353	INACTIVE	DO-TERM	230
354	INACTIVE	DO-TERM	240
355	INACTIVE	DO-TERM	239
356	INACTIVE	DO-TERM	249
357	INACTIVE	DO-TERM	248
360	INACTIVE	DO-TERM	221
370	2112B	DO-TERM	261
371	2147B	DO-TERM	268
372	INACTIVE	DO-TERM	275
373	INACTIVE	DO-TERM	274
375	INACTIVE	DO-TERM	260
376	INACTIVE	DO-TERM	267
380	2235B	DO-TERM	282
381	2272B	DO-TERM	289
382	INACTIVE	DO-TERM	296
383	INACTIVE	DO-TERM	295
385	INACTIVE	DO-TERM	281
386	INACTIVE	DO-TERM	288
390	2360B	DO-TERM	303
391	2415B	DO-TERM	310

```
1         PROGRAM TEZ1       74/835   OPT=0,ROUND= A/ S/ M/-D,-DS    FTN 5.1+670       90/10/23. 20.27.38      PAGE    12
  -LABEL-ADDRESS-----PROPERTIES----DEF        -LABEL-ADDRESS-----PROPERTIES----DEF      -LABEL-ADDRESS-----PROPERTIES----DEF

    392  INACTIVE   DO-TERM      317          550  INACTIVE   DO-TERM      412          1010  2160B                    271
    393  INACTIVE   DO-TERM      316          560  INACTIVE   DO-TERM      422          1020  2207B                    276
    395  INACTIVE   DO-TERM      302          570  INACTIVE   DO-TERM      421          1417  4063B                    516
    396  INACTIVE   DO-TERM      309          580  INACTIVE   DO-TERM      428          1500  2217B                    279
    400  INACTIVE   DO-TERM      321          590  INACTIVE   DO-TERM      427          1505  2253B                    286
    410  INACTIVE   DO-TERM      324          600  INACTIVE   DO-TERM      435          1510  2303B                    292
    420  INACTIVE   DO-TERM      327          610  INACTIVE   DO-TERM      438          1520  2332B                    297
    430  INACTIVE   DO-TERM      332          620  INACTIVE   DO-TERM      441          2000  2342B                    300
    440  INACTIVE   DO-TERM      339          630  INACTIVE   DO-TERM      482          2005  2376B                    307
    441  INACTIVE   DO-TERM      346          640  INACTIVE   DO-TERM      481          2010  2426B                    313
    442  INACTIVE   DO-TERM      345          641  INACTIVE   DO-TERM      480          2020  2455B                    319
    443  INACTIVE   DO-TERM      353          650  INACTIVE   DO-TERM      508          3000  2775B                    366
    444  INACTIVE   DO-TERM      352          651  INACTIVE   DO-TERM      515          4500  3033B                    381
    450  INACTIVE   DO-TERM      338          652  INACTIVE   DO-TERM      514          4650  3024B                    377
    460  INACTIVE   DO-TERM      358          660  INACTIVE   DO-TERM      507          5000  3770B                    500
    470  INACTIVE   DO-TERM      361          700  INACTIVE   DO-TERM      489          6500  4263B                    547
    480  INACTIVE   DO-TERM      364          710  INACTIVE   DO-TERM      488          7500  3142B                    396
    490  INACTIVE   DO-TERM      384          711  INACTIVE   DO-TERM      487          7650  4234B                    536
    500  INACTIVE   DO-TERM      387          720  INACTIVE   DO-TERM      496          8000  3340B                    432
    510  INACTIVE   DO-TERM      390          721  INACTIVE   DO-TERM      494          8500  3447B                    443
    520  INACTIVE   DO-TERM      404          730  INACTIVE   DO-TERM      495          9000  3476B                    454
    530  INACTIVE   DO-TERM      403         1000   2074B                  258          9500  4210B                    526
    540  INACTIVE   DO-TERM      413         1005   2130B                  265

  --ENTRY POINTS--(LO=A)
   -NAME---ADDRESS--ARGS---

    TEZ1        203     0

  --I/O UNITS--(LO=A)
   -NAME--- PROPERTIES-------------

    TAPE6    FMT/SEQ
    TAPE7    FMT/SEQ

  --STATISTICS--

    PROGRAM-UNIT LENGTH          133750B =  47080
    CM STORAGE USED               67500B =  28480
    COMPILE TIME                  5.350 SECONDS
```

0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0
1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0
1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

EXP.ASS.JOBS WEEKS

0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0

0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0

1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ASS.JOBS WEEKS

0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0

0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
0	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0

0	0	0
0	0	1
0	0	0
0	1	0
0	1	0
0	1	0
0	0	0
0	0	0
0	0	0
1	0	0
1	0	0
1	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0

CONFIGURATION OF THE COMPUTER SYSTEM USED

Hardware:

Network Operating System (NOS) Version 2.0

Batch processing on Control Data Cyber 170 Computer System.

524 K Memory

2*Line and 8*Matrix Control Data Cyber 9334 Printer.

2*Magnetic Disk Drive 883-6

2*Magnetic Tape Drive 679-5

Software:

Fortran 5 Compiler

BIBLIOGRAPHY

1.Balas,E.,"An Additive Algorithm for Solving Linear Programs with Zero-One Variables",Operations Research,Vol.13,pp.517-549,1965.

2.Geoffrion,A.M.,"Integer Programming by Implicit Enumeration and Balas' Method",SIAM Review,Vol.7,pp.178-190,1967.

3.Glover,F.,"A Multiphase-Dual Algorithm for the Zero-One Integer Programming Problem",Operations Research,Vol.113,pp.879-919,1965.

4.Glover,F. and Zionts,S.,"A Note on the Additive Algorithm of Balas", Operations Research,Vol.13,pp.546-549,1965.

5.Fleischmann,B.,"Computational Experience with the Algorithm of Balas", Operations Research,Vol.15,pp.153-155,1967.

6.Petersen,C.C.,"Computational Experience with Variants of the Balas Algorithm Applied to the Selection of Research and Development Projects", Management Science,Vol.13,pp.736-750,1967.

7.Tuan,P.N.,"A Flexible Tree Search Method for Integer Programming Problems",Operations Research,Vol.19,pp.115-119,1971.

8.Donald R.Plane and Claude McMillan,Jr., Discrete Optimization - Integer Programming and Network Analysis for Management Decisions, Prentice-Hall,Inc.,1971