T-1607

INITIALIZING IMPLICIT ENUMERATION
WITH A
SOLUTION OBTAINED FROM THE HEURISTIC
OF
SENJU-TOYODA

BY

ROBERT L. JOHNSON

A thesis submitted to the Faculty and the Board of Trustees of the Colorado School of Mines in partial fulfillment of the requirements for the degree of Master of Science in Mathematics.

Signed: Robert L. Johnson
Robert L. Johnson

Golden, Colorado

Date: Dec. 5, 1973

Approved: Robert E. D. Woolsey
Dr. R. E. D. Woolsey
Thesis Advisor

Robert A. Walsh
Dr. Robert A. Walsh, Head
Department of Mathematics

Golden, Colorado

Date: Dec 10, 1973

TABLE OF CONTENTS

ABSTRACT

Linear binary-integer problems are problems whose variables are either 0 or 1; each constraint and objective function must be linear. They are frequently encountered in capital budgeting problems, assignment problems, and allocation problems.

There is no simple technique that guarantees an optimum solution for linear binary-integer problems. However, there are both rigorous and heuristic methods used in solving problems of this type. Three rigorous techniques are explicit enumeration, implicit enumeration, and branch-and-bound procedures. S. Senju and Y. Toyoda have developed a projection gradient heuristic that compares the extent to which each variable satisfies the linear constraints and the objective function. The method requires less computer time than rigorous techniques and provides optimum or near optimum results.

To investigate the effectiveness of initializing implicit enumeration with the solution from Senju-Toyoda, three steps are required. First, the method of Senju-Toyoda is run to obtain a solution, which is not necessarily optimum. Then the implicit enumeration routine is run to obtain an optimum solution. Finally, the solution obtained from Senju-Toyoda provides an initial value of the objective function to an implicit enumeration routine. The solution times are then compared and discussed.

INTRODUCTION

Businessmen often encounter the problem of deciding which projects (activities) should be attempted, given that they have several possible choices. Generally, it is not possible to do every project because there are limitations for manpower, money, material, and time. The managers must decide which projects to perform and which to discard. In his elimination process, the manager attempts to optimize (maximize or minimize) the expected benefit (profit, time, utility, etc.) of all the projects performed. Also, he must stay within resource limitations.

The purpose of this thesis is to compare computational results of two different methods for solving problems of this type (linear binary-integer), and to determine if a good initial solution enhances an implicit enumeration scheme. The first technique is that of Senju-Toyoda (1) a projection gradient method. The second is implicit enumeration (2), which involves examining a few solutions explicitly and eliminating those which are infeasible or which do not improve the present value of the objective function (ZBAR). The procedure is repeated until all solutions have been examined (either explicitly or implicitly). Only solutions that improve the value of the objective function are candidates for the optimum solution. A "very good" starting solution to the implicit enumeration routine, therefore, should provide a good (but not necessarily optimal) value of the objective function, which in turn should require fewer iterations for the enumeration technique. The heuristic method of Senju-Toyoda, which is much faster than implicit enumeration, is used to obtain such a starting solution. To compare the benefit of using a "good" starting solution, each sample problem is solved using implicit enumeration with and without the starting solution obtained from Senju-Toyoda. Computer results indicate that the projection gradient method usually obtains very good solutions (optimal or nearly optimal) with considerably less time than the implicit enumeration scheme. These results are displayed in Table 1.

Some applications of binary-integer programming problems are: capital budgeting, allocation, assembly line balancing, the location problem, the traveling salesman problem, and the knapsack problem (see (3) and (4)). Capital budgeting problems will be the primary concern in this thesis. It should be noted that some of these problems (for instance, the assignment problem) can be solved more adequately by "specialized" algorithms (see (5)). See Appendix C for examples and formulations of some of these problems.

The main objective in capital budgeting is to perform the activities that will give the greatest reward or profit and stay within resource limitations. This is important in today's businesses because resources are becoming less plentiful and more costly. Many industries that are plagued with lack of resources could probably benefit from capital budgeting information in their decision-making processes.

In general, the above-mentioned problem types, and their formulations, are important because they are concerned with using the available resources most efficiently while maximizing a rate of return (utility). With the inevitable scarcity of many important resources, businesses must consider employing optimization techniques in their daily decision policies. Hopefully, the results of this thesis will provide an insight for using a heuristic for solving some of these problems.

REVIEW OF LITERATURE

The purpose of this section is to present some background information related to linear zero-one problems. There are several techniques that can be used to solve these problems; some of these are explicit enumeration, implicit enumeration, branch-and-bound procedures, and heuristics. Implicit enumeration algorithms have been advanced by Balas (2), Geoffrion (6), and Glover (7); Lawler (8) has developed a branch-and-bound procedure.

Explicit enumeration involves examining 2^n solutions (where n = number of variables) directly, eliminating the infeasible ones and finding the optimum of the remaining solutions. Balas' implicit enumeration scheme partitions the variables into three classes: 1) those variables set to 0, 2) those variables set to 1, and 3) those which are free to be either 0 or 1. A few solutions are examined explicitly and infeasible solutions or ones that do not improve the value of the objective function are eliminated The process is repeated until all solutions have been examined. Hopefully, all solutions are not considered explicitly, although this is not guaranteed. This factor is an important advantage of implicit enumeration over explicit enumeration. The original formulation by Balas required severe computer storage for large problems (more than 20 variables), since all tentative solutions were stored. Also, problems requiring many iterations (say 1000) were very time consuming.

To alleviate storage and computer-time problems, Glover (7) and Geoffrion (6) applied a backtracking procedure in which only the variable subscripts were needed. A positive subscript in the set indicated that the corresponding variable was set equal to 1. A negative subscript indicated the variable was set to 0. All other variables were termed "free" variables because they were either 0 or 1.

In Glover's formulation, when a solution is eliminated because of infeasibility, the rightmost positive element in the set is negated and any elements to the right are removed from the set. The search is complete when all elements are negative. The reformulation seems to have considerably reduced both storage and computer time requirements (see (9)).

Branch-and-bound procedures (8 and 10) involve searching a branch of a "solution tree" until the solution becomes infeasible, the end of the branch is reached, or no solution on the branch improves the value of the objective function. In any of these cases, the routine considers another branch and continues the examination process until all branches have been investigated. Note that 2^n solutions are evaluated (where n = number of variables), perhaps not all of them explicitly.

The method of generalized Lagrange multipliers by Everett (11) can also be used to solve binary-integer problems. The algorithm attempts to reduce the original problem to an unconstrained optimization problem with the aid of Lagrange multipliers. Wiley (12) provided computational results for this technique.

The only heuristic approach considered is the projection gradient technique by Senju and Toyoda (1). Although this technique does not guarantee an optimal solution, it provides a nearly optimal solution very quickly. As presented later in this thesis, solution times are generally very much less than for the implicit enumeration scheme.

THE LINEAR BINARY-INTEGER PROBLEM

In general, the binary-integer problem may be formulated as follows:

Maximize $Z(X) = \sum_{j}^{N} C_j X_j$

Subject to $W(X)i = \sum_{j}^{N} A_{ij} X_j \leq B_i$ for all $i = 1,2,\ldots,M$

$X_j = 0$ or 1

where C_j = The return from doing project j

A_{ij} = The amount of resource i necessary to do project j

B_i = The amount of resource i available

M = The number of resources

N = The number of projects

$Z(X)$ = The sum of the returns of the activities to be performed

$W(X)$ = The amount of resource i required to do all of the selected projects.

For each solution the interpretation is:

$X_j = 1$ if project j is to be done and 0 otherwise.

The method of Senju-Toyoda requires this format along with the restriction that C_j, A_{ij}, and B_i are each nonnegative. Since these parameters are defined as above, there should be no difficulty encountered with this restriction for this particular application. However, difficulties may arise for problems not conforming to this restriction. Capital budgeting problems are very well suited for the format of Senju-Toyoda.

To convert the parameters defined above to the general format of implicit enumeration (developed by Balas), let $X_j = 1 - X_j'$.

Then,

$$\text{Maximize } Z(x) = \sum_{j}^{N} C_jX_j \quad \text{becomes}$$

$$\text{Maximize } Z(x) = \sum_{j}^{N} C_j(1-X_j') = \sum_{j}^{N} C_j - \sum_{j}^{N} C_jX_j'$$

and

$$\text{Maximize} \quad \sum_{j}^{N} C_j - \sum_{j}^{N} C_jX_j' = \text{Min} \sum_{j}^{N} C_jX_j' - \sum_{j}^{N} C_j$$

since the variables which maximize $f(x_1, x_2,\ldots,x_n)$ also minimize $-f(x_1, x_2,\ldots,x_n)$. Therefore,

$$\text{Maximize} \quad \sum_{j}^{N} C_jX_j = \text{Min} \sum_{j}^{N} C_jX_j' - \sum_{j}^{N} C_j$$

For each constraint, there is a similar transformation:

$$W(x)_i = \sum_{j}^{N} A_{ij}X_j \leq B_i \text{ becomes}$$

$$\sum_{j}^{N} A_{ij} (1-X_j') \leq B_i$$

or

$$\sum_{j}^{N} A_{ij} - \sum_{j}^{N} A_{ij}X_j' \leq B_i.$$

Now multiply by -1 so that the inequality is properly oriented:

$$-\sum_{j}^{N} A_{ij} + \sum_{j}^{N} A_{ij}X_j' \geq -B_i$$

or

$$\sum_{j}^{N} A_{ij}X_j' \geq \sum_{j}^{N} A_{ij} - B_i$$

For more information on this matter, see Hiller and Lieberman, page 151.

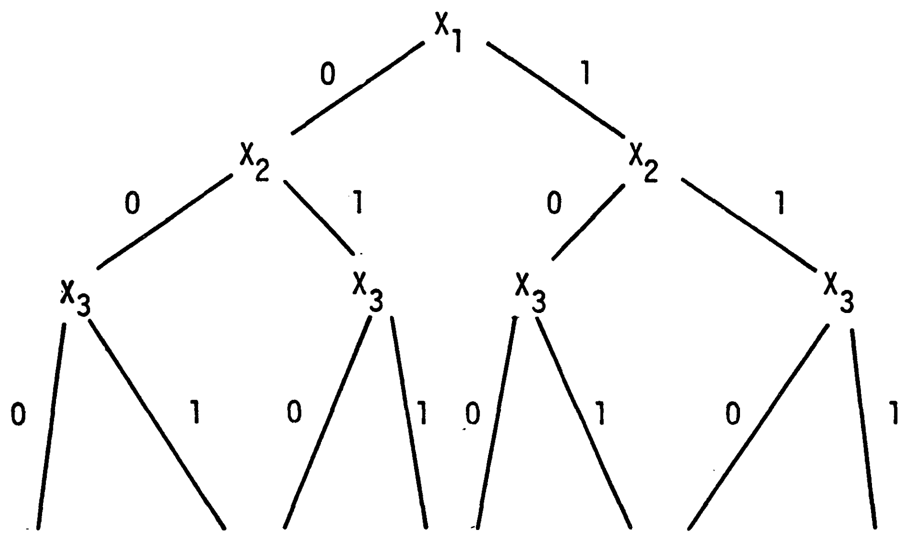
Having made the necessary changes, the general formulation is:

$$\text{Minimize} \quad F(x) = \sum_{j}^{N} C_j X_j' - \sum_{j}^{N} C_j$$

$$\text{Subject to } G(x) = \sum_{j}^{N} A_{ij} X_j' \geq \sum_{j}^{N} A_{ij} - B_i \qquad i = 1,2,\ldots,M$$

$$X_j' = 0 \text{ or } 1.$$

Since the problems we are concerned with in this paper (capital budgeting problems) have variables whose values are restricted to 0 or 1, linear programming (which requires continuous variables) is not very suitable for solving them. However, as stated previously, such techniques as explicit enumeration, implicit enumeration, and branch-and-bound procedures provide optimal solutions. While explicit enumeration must investigate all solutions (2^n of them) directly, implicit enumeration, hopefully, must consider only a few. Branch-and-bound procedures can be best visualized with the aid of a "tree" diagram. An example consisting of three variables is presented on the following page.



A Three-Variable Branch-and-Bound Tree

The 0's and 1's indicate the choices of values for each variable. The routine follows a branch in the tree until no solution on the branch improves the present optimum, the solution becomes infeasible, or the branch terminates. At this point, the routine considers another branch and terminates if all branches have been examined.

The last-mentioned technique for solving linear binary problems is the projection gradient heuristic by S. Senju and Y. Toyoda. Their algorithm does not guarantee finding an optimum solution; even if one exists. However, the heuristic provides near optimum results with less computer time than rigorous techniques. It will be used to provide a starting solution to the implicit enumeration scheme to investigate if a good starting solution enhances the implicit enumeration scheme.

METHOD OF SENJU-TOYODA

In this section the algorithm for the projection gradient scheme of Senju and Toyoda is developed. All necessary terms are defined and a simple example is presented.

Define the following:

1. $j = 1, 2, \ldots, N$ activities
2. $\underline{P}_j$ = n-vector giving the relationship between all n processes/ resources
3. $\underline{R}$ = vector sum of all $\underline{P}_j$; $\underline{R} = \sum_j \underline{P}_j$
4. $\underline{L}$ = limitation vector; its components are the process/ resource limitations
5. $\underline{S}$ = surplus vector; $\underline{S} = \underline{R} - \underline{L}$
6. $\underline{U} = \underline{S}/|\underline{S}|$ = unit vector parallel to $\underline{S}$ and having the same sense
7. $\underline{U}_j$= projection of the vector $-\underline{P}_j$ onto the vector $-\underline{U}$
8. $\underline{U}_j = (-\underline{P}_j) * (-\underline{U}_j)$, where * indicates scalar product
9. C_j= "profit" (or benefit) of the jth activity
10. $EG_j = C_j/U_j$ = Effective Gradient of the vector $\underline{P}_j$; the Effective Gradient is the profit C_j divided by its projection U_j
11. A_{ij} = coefficient matrix for constraints.

Since it is not generally possible to perform every activity in the objective function (due to resource limitation, etc.), we must select the activities that satisfy all of the constraints as well as the problem objective (maximize or minimize). The gradient projection algorithm presented here provides a heuristic approach for accomplishing these goals.

Having defined the necessary terms, an intuitive technique is represented with the aid of the following graph which shows the relationship between two processes.

Determination of $\underline{R}$ for Two Processes

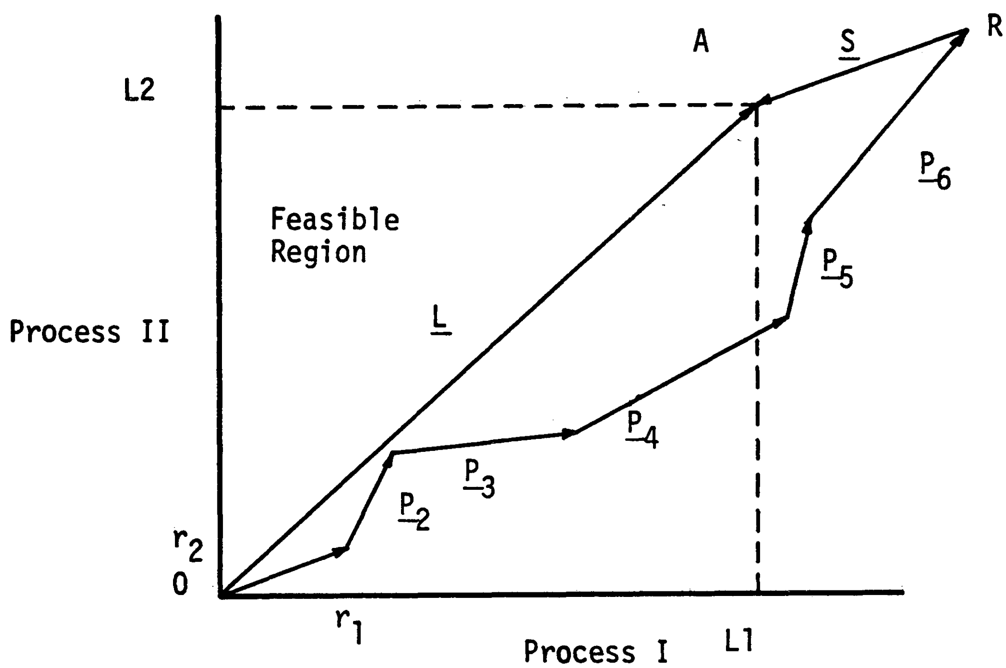


Figure 1

L_1 and L_2 are the resource limitations of process I and II, respectively. The resultant vector, $\underline{R}$, is the sum of the individual vectors $\underline{P}_j$. Vector $\underline{P}_1$ shows that r_1 units for process one is used with r_2 units for process two if activity one is performed. $\underline{S}$ is simply $\underline{R} - \underline{L}$ where $\underline{L}$ is the vector sum of the limitations, i.e. $|\underline{L}| = \sum_j L_j$.

The vector $\underline{S}$ provides two important things: the <u>direction</u> and <u>extent</u> for removing the P-vectors. It is $\underline{S}$ upon which the gradients of each $\underline{P}_j$ is projected. See the following figure.

Projection of P onto S

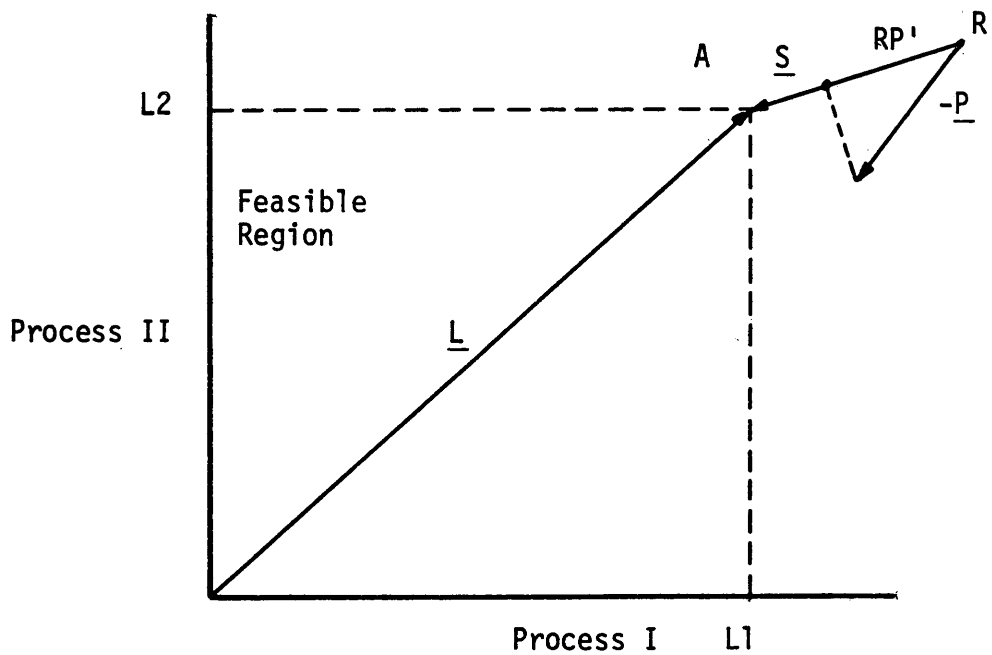


Figure 2

It should be noted that the order for adding the P-vectors is random. Therefore, to return to the feasible region, any vector can be removed, and the algorithm provides a direction and extent. More will be said about this later.

When any vector is omitted (say $\underline{P_6}$ in Figure 1), the point R is moved along the vector $(-\underline{P_6})$ to the point RP (see Figure 2). This move causes a reduction in profit (or whatever the objective function is concerned with) by the amount associated with performing activity 6. The contribution from $\underline{P_6}$ from moving toward point A is the projection of $\underline{P_6}$ onto $\underline{S}$. Naturally, the object is to eliminate activities whose profits are relatively small when compared with their projected length on $\underline{S}$.

In practice, what needs to be compared is the effective gradients of all the proposed activities. Once the gradients have been calculated, consider eliminating the ones that provide the least gain or profit--these are designated by small (relatively) effective gradients. The step-wise procedure for finding EG_j (effective gradient) is:

1. $\underline{U} = \underline{S}/|\underline{S}| = \left(\frac{S_1}{(S_1^2 + S_2^2 + \ldots + S_n^2)^{1/2}}, \frac{S_2}{(S_1^2 + S_2^2 + \ldots + S_n^2)^{1/2}}, \ldots, \frac{S_n}{(S_1^2 + S_2^2 + \ldots + S_n^2)^{1/2}} \right)$

where S_1, S_2, ... , S_n are components to $\underline{S}$

2. $U_j = (-\underline{P}_j) * (-\underline{U})$

3. $EG_j = C_j/U_j$

Note that each component of the vector $\underline{U}$ has a common denominator $(S_1^2 + S_2^2 + \ldots + S_n^2)^{1/2}$ This gives a common factor in all EG_j, i.e.

$$EG_1 = a_1 (S_1^2 + S_2^2 + \ldots + S_n^2)^{1/2}$$

$$EG_2 = a_2 (S_1^2 + S_2^2 + \ldots + S_n^2)^{1/2}$$

$$EG_j = a_j (S_1^2 + S_2^2 + \ldots + S_n^2)^{1/2}$$

and therefore the factor $(S_1^2 + S_2^2 + \ldots + S_n^2)^{1/2}$ may be omitted from the calculations.

The effective gradients are compared by comparing the values a_1, $a_2, \ldots, a_j$ and the projects with smaller effective gradients are removed until feasibility is reached, at which time each project is checked to see if it can be added back into the solution (and still maintain feasibility).

Perhaps a summary of the procedure is useful:

Step 1: Put the problem in the proper format. Initialize by setting all variables to 1.

Step 2: For each row, find the total resource requirement-- $R_i = \sum_j A_{ij}$ for each i. For each row find the difference between the supply and demand of resource i -- $S_i = R_i - B_i$.

Step 3: For each project (column) find the effective gradient.

Step 4: Find the project with the least effective gradient and remove it from consideration and subtract its A_{ij} from the total requirement for each row (in short, subtract A_{ij} from S_i).

Step 5: Is the solution feasible? If yes, go to step 6. If no, go to step 4.

Step 6: Decide if there are any projects that can be added back and still maintain feasibility. Start with the project with the greatest payoff that was removed from consideration and continue until no more projects can be added without becoming infeasible.

The sketch on the following page shows how the gradient vectors are added and how they can be removed to get back into the feasible region. In trying to obtain feasibility, we choose the vectors that will best "fit" the vector $\underline{S}$. A simple example is used to explain the procedure and the figure.

ADDITION OF P-VECTORS AND HOW TO OBTAIN FEASIBILITY

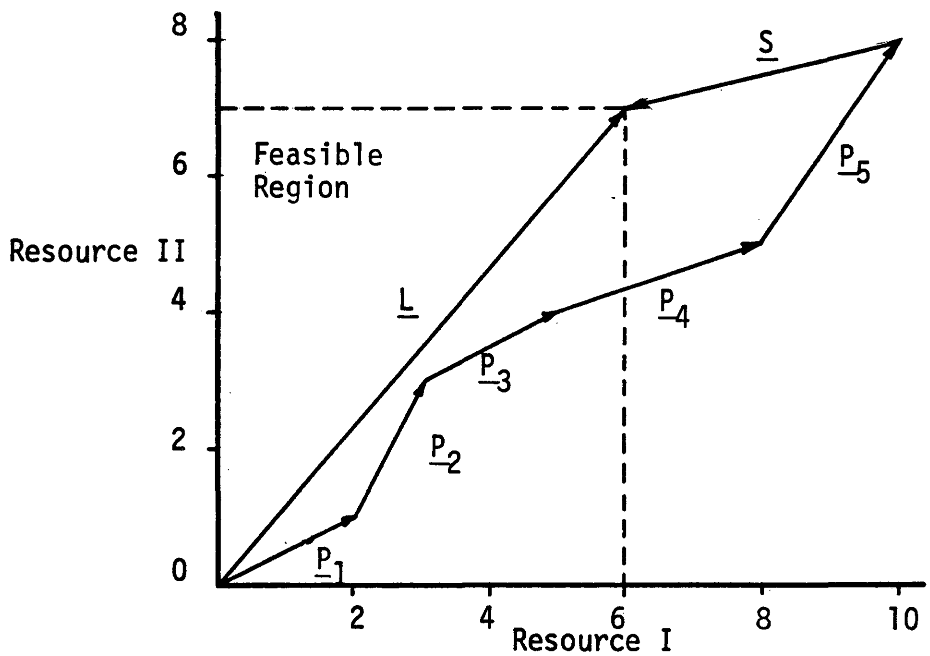


Figure 3

Consider the following five-variable example, in which $\underline{P}_j = (r_{1j}, r_{2j})$.

$$\text{Maximize} \quad X_1 + 2X_2 + 2X_3 + X_4 + 4X_5$$

$$\text{S. T.} \quad 2X_1 + X_2 + 2X_3 + 3X_4 + 2X_5 \leq 7 \text{ (Resource I)}$$

$$X_1 + 2X_2 + X_3 + X_4 + 3X_5 \leq 6 \text{ (Resource II)}$$

$$X_j = 0 \text{ or } 1 \text{ for all } j.$$

Then

$$X_1 = 1 \rightarrow \begin{matrix} r_{11} = 2 \\ r_{12} = 1 \end{matrix}, \quad X_2 = 1 \rightarrow \begin{matrix} r_{21} = 1 \\ r_{22} = 2 \end{matrix}, \quad X_3 = 1 \rightarrow \begin{matrix} r_{31} = 2 \\ r_{32} = 1 \end{matrix},$$

$$X_4 = 1 \rightarrow \begin{matrix} r_{41} = 3 \\ r_{42} = 1 \end{matrix}, \quad X_5 = 1 \rightarrow \begin{matrix} r_{51} = 2 \\ r_{52} = 3 \end{matrix}$$

($\rightarrow$ means "implies")

$\underline{R} = \left\{ \sum_i r_{i1}, \ \sum_j r_{1j} \right\} = (10,8)$, where i = index for the activity number and j = index for the resource number (1 or 2). Now find each $R_i = \sum_j^N A_{ij}$: $R_1 = 10$ and $R_2 = 8$ (note that these are components of $\underline{R}$).

The difference between supply and demand is $S_i = R_i - L_i$:

$S_1 = 10-7 = 3$ and $S_2 = 8-6 = 2$, or $\underline{S} = \underline{R} - \underline{L} = (R_1-L_1,\ R_2-L_2) = (10-7, 8-6)$.

Next find

$$\underline{U} = \underline{S}/\ |\underline{S}| = [S_1/\ (S_1^{\ 2} + S_2^{\ 2})^{1/2},\ S_2/(S_1^{\ 2} + S_2^{\ 2})^{1/2}]$$

$$\underline{U} = (3/(9 + 4)^{1/2},\ 2/(9 + 4)^{1/2})$$

and $U_1 = \underline{P_1}*\underline{U} = (2,1) * (3/(13)^{1/2},\ 2/(13)^{1/2}) = (6 + 2)/(13)^{1/2} = 8/(13)^{1/2}$

$U_2 = \underline{P_2}*\underline{U} = (1,2) *$ " $= (3 + 2)/(13)^{1/2} = 5/(13)^{1/2}$

$U_3 = \underline{P_3}*\underline{U} = (2,1) *$ " $= (6 + 2)/(13)^{1/2} = 8/(13)^{1/2}$

$U_4 = \underline{P_4}*\underline{U} = (3,1) *$ " $= (9 + 2)/(13)^{1/2} = 11/(13)^{1/2}$

$U_5 = \underline{P_5}*\underline{U} = (2,3) *$ " $= (6 + 6)/(13)^{1/2} = 12/(13)^{1/2}$

If the coefficients in the objective function are interpreted as "profit" values, the effective gradients are given by $EG_j = C_j/U_j$:

$EG_1 = (1/8\ )x(13)^{1/2}$

$EG_2 = (2/5\ )x(13)^{1/2}$

$EG_3 = (2/8\ )x(13)^{1/2}$

$EG_4 = (1/11)x(13)^{1/2}$

$EG_5 = (4/12)x(13)^{1/2}$.

$$EG_j = \frac{C_j x\ (S_1^{\ 2}+S_2^{\ 2}+\ldots+S_n^{\ 2})^{1/2}}{\sum_i^M p_i\ s_i}$$

where p_i = ith component of $\underline{P}$ and

s_i = ith component of $\underline{S}$.

Notice that $(13)^{1/2}$ is common in each EG_j. Therefore, to compare the effective gradients, one needs only to compute/compare the factor preceding $(13)^{1/2}$. Then tabulate the effective gradients in increasing order:

$EG_4 = 1/11$

$EG_1 = 1/8$

$EG_3 = 1/4$

$EG_5 = 1/3$

$EG_2 = 2/5.$

Now remove proposed activities, starting with the ones having smaller effective gradients, until the resultant vector $\underline{R}$ enters the feasible region. At this time, check to see if any of the removed activities can be reinserted into the solution set. If this is possible, the total profit is increased.

Since EG_4 is smallest, remove variable X_4 (let $X_4 = 0$). The surplus values are:

$S_1 - A_{14} = 3 - 3 = 0 = S_1'$

$S_2 - A_{24} = 2 - 1 = 1 = S_2'$

and since $S_2' > 0$, remove EG_1(let $X_1 = 0$). The new surpluses are:

$S_1' - A_{11} = 0 - 2 = -2 = S_1''$

$S_2' - A_{12} = 1 - 1 = 0 = S_2''.$

The resultant vector finally enters the feasible region because there are no surpluses. Therefore, consider adding either X_1 or X_4 back into the solution. Consider adding X_4 (since C_4 is greater than C_1):

$S_1'' + A_{14} = -2 + 3 = 1$

$S_2'' + A_{24} = 0 + 1 = 1.$

Since there are positive surpluses, $\underline{R}$ is outside the feasible region and X_4 cannot be added back into the solution. This is also true for x_1. Therefore, the solution is $X_1 = X_4 = 0$, $X_2 = X_3 = X_5 = 1$. The corresponding value of the objective function (ZBAR) is 8.

In conclusion, it is suggested that the S-vector is "occasionally" recomputed. "Occasionally" may mean after a project is eliminated or it may be after a fixed number of activities are removed (say five). This should provide more accuracy in determining which vector $\underline{P}$ reduces feasibility by the largest amount.

METHOD OF IMPLICIT ENUMERATION

Implicit enumeration is a branch-and-bound procedure for solving binary-integer problems. The algorithm involves examining a few solutions completely and eliminating those that are infeasible or that do not improve the current value of the objective function. Additional solutions are examined explicitly until all solutions have been considered. The main difference between explicit enumeration and implicit enumeration is that explicit enumeration considers all solutions whereas implicit enumeration (generally) must examine only a few solutions directly.

To outline the routine we must define the following terms and expressions:

1. S = the partial solution vector which contains the subscripts of the variables which have been given specific values--0 or 1. The positive subscripts refer to the variables set to 1; the negative subscripts refer to the variables set to 0. Each variable added to S is placed to the right of the rightmost element already in S. This insures that, while backtracking, we consider only solutions not previously examined.
2. free variables = those variables not in S. Their values have not been specified as either 0 or 1.
3. V = the set of constraints violated when the partial solution S is considered and the free variables are set equal to 0.
4. ZBAR = current value of the objective function.

5. XBAR = $(x_1, x_2, \ldots, x_n)$ = the current solution.
6. f_p = the value of the objective function obtained by setting to zero all of the free variables.
7. T = the set containing each free variable which has
   i) an objective function coefficient less than the limit= ZBAR $\div f_p$ <u>and</u>
   ii) a positive coefficient in some constraint in V.
8. complement of a partial solution -- all possible solutions associated with the partial solution. That is, if S = (1,2), then the completions of S are all those solutions for which $X_1 = X_2 = 1$.
9. backtrack -- the procedure for going "backwards" in the solution network. It is used to examine all possible solutions for a partial solution S.

Having defined the necessary terms, the <u>general</u> procedure for solving 0-1 problems via implicit enumeration can be outlined. First, start with the partial solution S empty -- S = $\emptyset$. Identify the set of violated constraints V and store in the set T those variables which cause infeasibility when equal to 1. If every constraint in V can be made feasible by adding (to the solution) variables in T, we add to S that variable in T which has the greatest coefficient sum in all of the constraints.

Continue this process until a partial solution has been implicitly enumerated. Then backtrack, complementing (negating) the rightmost element in S which has not been complemented, and drop any elements in S to the right of the variable currently complemented and begin to consider the partial solution.

The step-by-step procedure is presented in the flow diagram, which is the last page in this section. Now consider a simple example.

$$\text{Minimize } F = 2\,X_1 + 3\,X_2 + X_3$$

$$\text{S. T.} \quad 4\,X_1 + X_2 - 2\,X_3 \geq 3 \qquad (1)$$

$$X_1 + 3\,X_2 - X_3 \geq 4 \qquad (2)$$

$$X_j = 0,1.$$

Let k denote the iteration number. $ZBAR_k = 2 + 3 + 1 = 6 = ZBAR_0$. If $S_0 = \emptyset$ is completed by setting the free variables to 0, then

	X_1	X_2	X_3	
(1)	4 (0) +	1 (0) -	2 (0)	$= 0 \not\geq 3$
(2)	2 (0) +	3 (0) -	1 (0)	$= 0 \not\geq 4$

and since both constraints are violated, $V_0 = (1,2)$. Since $V_0 \neq \emptyset$ and $S = \emptyset$. $f_p = 0$ and the limit = ZBAR - f_p = 6. The variables with positive coefficients are: constraint (1) X_1, X_2

constraint (2) X_1, X_2.

Therefore, T = (1,2). To obtain feasibility, one or more of these variables must be raised to 1.

For constraint (1) $X_1 = X_2 = 1;\ X_3 = 0$

$$4\,(1) + 1\,(1) - 2\,(0) = 5 \geq 3 \text{ --OK}$$

constraint (2) $X_1 = X_2 = 1;\ X_3 = 0$

$$2\,(1) + 3\,(1) - 1\,(0) = 5 \geq 4 \text{ --OK}$$

Then find the coefficient sums of the variables in T and raise to 1 the variable with the greatest sum:

for X_1 : $4 + 1 = 5$ *

for X_2 : $1 + 3 = 4$.

Now set $X_1 = 1$, hence $S_1 = (1)$. The variables not in S are the free variables. When S is completed by setting all free variables to 0, constraint 2 is violated; thus $V = (2)$.

Preforming another iteration shows that X_2 must be raised to 1, which allows for both constraints to be satisfied, i.e. $V = \emptyset$. The partial solution is $S_2 = (1,2)$, indicating that $X_1 = X_2 = 1$.

Since all constraints are feasible for $X_1 = X_2 = 1$; $X_3 = 0$, the solution vector is XBAR = (1,1,0) and ZBAR = 2 (1) + 3 (1) + 1 (0) = 5. Note that any solution containing $X_3 = 1$ causes constraint 2 to be infeasible. Therefore, the solution (1,1,0) is optimal.

For a more detailed description of this algorithm, see the flow diagram and the additional problem in the appendix.

In summary, implicit enumeration keeps track of all the solutions that are infeasible or do not improve the value of the objective function. For example, the constraint $4 X_1 - 2 X_2 + X_3 - 5 X_4 \geq 5$ is <u>always</u> violated when $X_1 = 0$. Therefore, every solution for which $X_1 = 0$ can be eliminated from consideration. Thus, all related solutions (with $S_1 = 0$) have been <u>implicitly</u> enumerated.

FLOW DIAGRAM FOR ZERO-ONE PROBLEMS
VIA IMPLICIT ENUMERATION

Step 0 : Formulate the problem as

Minimize $F = \sum_{j=1}^{N} C_j X_j$

Subject to $g_i \geq 0 \qquad i = 1,2,\ldots,M$

$X_j = 0, 1$

where g_i are the constraints

Step 1 : If the solution "all X's = 0" is feasible, the problem is solved. Otherwise, set ZBAR = f evaluated at "all X's = 1", or the best known upper bound for the objective function, whichever is smaller. (ZBAR = min (F $(x_1=1, x_2=1,\ldots,x_n=1)$, known bound)). Find the coefficient sum for each variable and set S = $\emptyset$ (null set).

Step 2 : Find V, the set of constraints violated when the partial solution S is completed by setting to zero all variables not in S.

Step 3 : If V is empty, go to Step 9. Otherwise, go to Step 4.

Step 4 : Find f_p, the value of f when S is completed by setting to zero all variables not in S. Set the objective function coefficient limit to ZBAR $-f_p$.

Step 5 : Store in T each variable not in S which has:

a) an objective function coefficient less than the limit ZBAR $-f_p$, and

b) a positive coefficient in some constraint in V

Step 6 : If T is empty, go to Step 11. Otherwise, go to Step 7.

Step 7 : Can every constraint in V be made feasible by adding only variables in T? If the answer is yes, go to Step 8. Otherwise, go to Step 11.

Step 8 : Add to S the variable in T with the greatest coefficient sum. Go to Step 2.

Step 9 : Complete the partial solution S by setting to zero all variables not in S. This completed solution becomes the incumbent solution XBAR and the value of the objective function at XBAR becomes the new value of ZBAR.

Step 10: Locate the rightmost positive element in S. Replace it with its negative (complement), and drop any elements to the right. Go to Step 2.

Step 11: Are all the elements in S negative? If the answer is yes, go to Step 12. Otherwise, go to Step 10.

Step 12: Terminate. The incumbent solution (if there is one) is an optimal solution. If there is no incumbent solution, then there is no feasible solution better the solution corresponding to the best known upper bound used in Step 1.

RESULTS

Table 1 shows the times necessary to run each technique on the CSM time-sharing PDP-10 computer. All times are in milliseconds (unless otherwise specified), and, unfortunately are somewhat proportional to the amount of "swapping" that is necessary in running the routines.

There are 16 problems presented, consisting of 10 to 39 variables and 1 to 36 constraints. The problems were selected because of their diversity in size and individual densities.

The first four problems represent a particular class, called knapsack problems. The remaining ones represent a more general class of capital-budgeting problems.

Solution times vary from 0 - 150 milliseconds for Senju-Toyoda; from 83 milliseconds to 2934 seconds for implicit enumeration without a "good" initial solution, and from 66 milliseconds to 2720 seconds for implicit enumeration with Senju-Toyoda. These variations in solution times are presented below:

METHOD	MIN TIME - MAX TIME
Senju-Toyoda	0 - 150 milliseconds
Implicit enumeration without Senju-Toyoda	83 milliseconds to 2934 seconds
Implicit enumeration with Senju-Toyoda	66 milliseconds to 2720 seconds

Fifteen of the problems required less than 65 milliseconds to run with Senju-Toyoda alone. Six problems required more than 2 seconds to run with the implicit enumeration routine (without Senju-Toyoda) -- two of these required over 30 seconds; one required more than 5 minutes; and one required nearly 50 minutes. The implicit enumeration routine

TABLE 1 -COMPARISON OF METHODS

				Senju-Toyoda			Implicit Enumeration w/out Senju-Toyoda			Implicit Enumeration with Senju-Toyoda		
	Prob*/ Source	Size *	Optimum	Solution Time**	Z	%OPT	Solution Time	Z	%OPT	Solution Time	Z	%OPT
1.	(14)	(10, 1)	58	17	52	89.7	200	58	100	184	58	100
2.	(14)	(10, 1)	68	0	67	98.5	217	68	100	167	68	100
3.	(14)	(10, 1)	62	0	52	84.9	217	62	100	150	62	100
4.	(14)	(10, 1)	52	0	52	100.0	284	52	100	183	52	100
5.	(12)	(4, 2)	7	17	7	100.0	83	7	100	66	7	100
6.	(12)	(4, 3)	8	17	8	100.0	100	8	100	67	8	100
7.	(15)	(4, 4)	7	0	7	100.0	116	7	100	83	7	100
8.	(9)	(6, 10)	3800	16	3800	100.0	184	3800	100	134	3800	100
9.	(15)	(10, 5)	2	17	2	100.0	333	2	100	100	2	100
10.	(15)	(10, 5)	5	0	5	100.0	3.5s	5	100	3.2s	5	100
11.	(9)	(10,10)	8706.1	17	8336.9	95.8	366	8706.1	100	333	8706.1	100
12.	(9)	(15,10)	4015	33	3760	93.6	2.5s	4015	100	2.3s	4015	100
13.	(9)	(20,10)	6120	33	6010	98.2	34.8s	6120	100	32.1s	6120	100
14.		(25,36)	19	150	19	100.0	32.8s	19	100	31.4s	19	100
15.	(9)	(28,10)	12,400	47	12,400	100.0	457.1s	12400	100	428.2s	12,400	100
16.		(39, 5)	10,618	63	10,279	96.7	2934.0s	10,618	100	2720.0s	10,618	100

*(N,M) N = Number of variables; M - number of constraints

** All times in milliseconds unless otherwise specified

s = seconds

with Senju-Toyoda had from 5.3% to 70% improvement over implicit enumeration considered alone. The average improvement was 20.4%. See Table 3.

The sample problems considered have densities that vary from 12 to 100 %. The following table exhibits the densities for the problems that are <u>not</u> 100 % dense.

TABLE 2 - PROBLEMS LESS THAN 100% DENSE

PROB #	SIZE	% DENSE
8	(6 , 10)	85.0
11	(10, 10)	99.0
12	(15, 10)	96.7
13	(20, 10)	94.5
14	(25, 36)	12.0
15	(28, 10)	94.3
16	(39, 5)	89.3

The average percent density is 91.9% (including all 16 problems). The standard deviation of the density is 21%. The average density, neglecting problem 14(which is 12% dense), is 96.6% and the standard deviation is 3.1%. The least-dense problem took Senju-Toyoda more than twice as long to run than the second most time-consuming problem. Implicit enumeration with Senju-Toyoda was consistently faster than implicit enumeration alone for all densities. The method of Senju-Toyoda obtained a solution less than 90% optimum for only two problems - and each of these consisted of only one constraint. Optimal solutions were obtained for one knapsack problem (problem 4), five small problems with 10 variables or less (problems 5 - 10), and two large problems having more than 20 variables (problems 14 and 15). Solutions providing better than 93% of the optimum (but suboptimal) are found in problems 2, 11, 12, 13, and 16. The average percent of optimum provided by Senju-Toyoda is 97.2%. The standard deviation is 4.3%. Note that optimal solutions were obtained over the entire range of densities.

The implicit enumeration scheme achieved optimality for all problems. Therefore, to determine how effectively Senju-Toyoda enhances the implicit enumeration routine, the solution time required without a "good" starting solution must be compared to the solution time required with a "good" starting solution. For every problem considered, implicit enumeration with Senju-Toyoda required less time than implicit enumeration without a "good" starting solution. On the average, implicit enumeration with Senju-Toyoda required 20% less computer time than implicit enumeration without a "good" starting solution. The standard deviation for this time difference is 15%. The following bargraph shows the relative differences in computer times for the three routines. (See page 29).

Perhaps it is useful to display the percent change in time for implicit enumeration with and without an initial solution from Senju-Toyoda. Let

I-E = implicit enumeration

S-T = Senju-Toyoda

I-E w/S-T = I-E with S-T

I-E w/out S-T = I-E without S-T.

Then the percent calculation is

$$X\% = \frac{(\text{I-E w/out S-T})-(\text{I-E w/S-T})}{(\text{I-E w/out S-T})} \times 100,$$

which is displayed in Table 3.

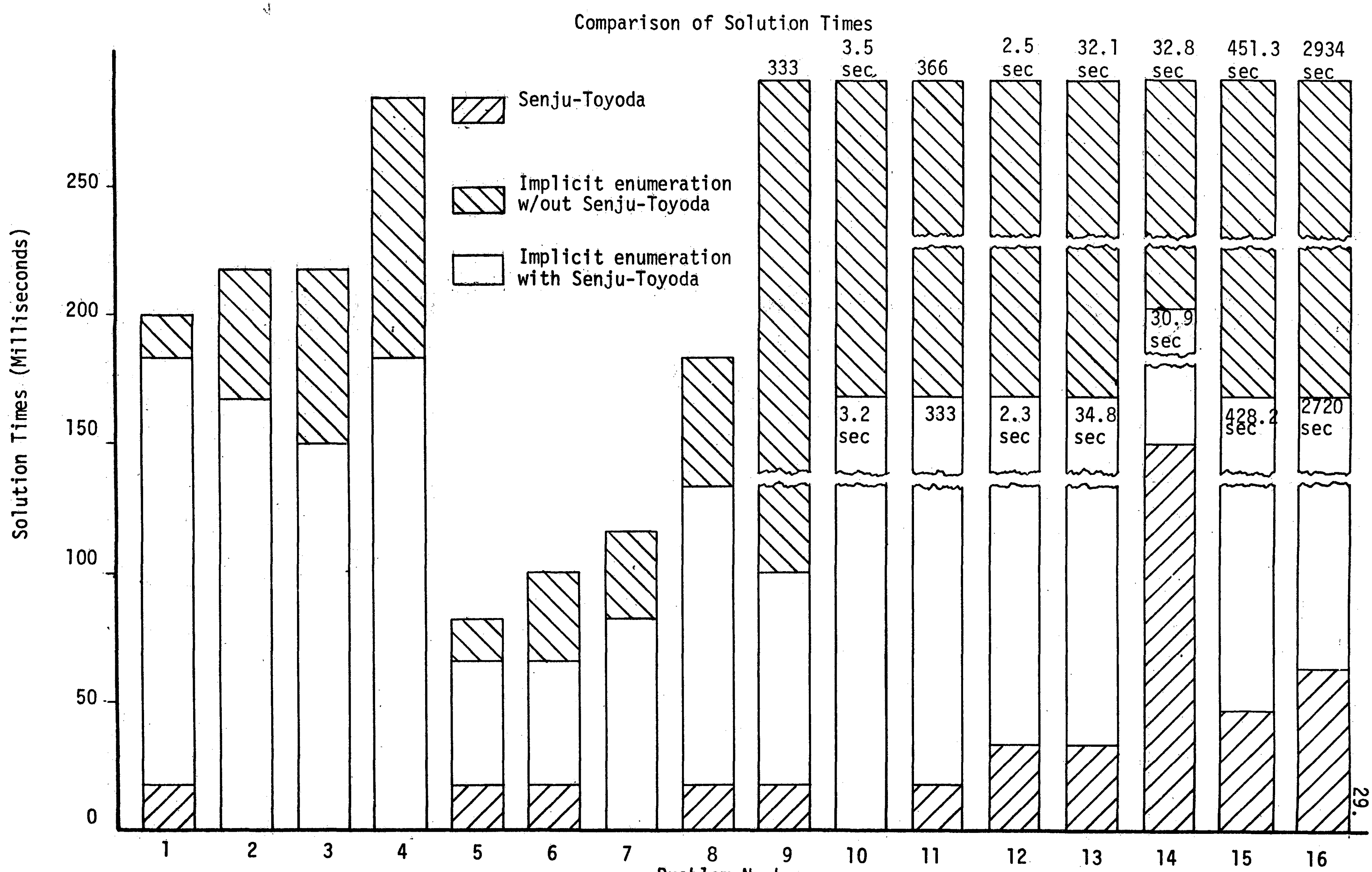

Comparison of Solution Times
Senju-Toyoda
Implicit enumeration w/out Senju-Toyoda
Implicit enumeration with Senju-Toyoda
Solution Times (Milliseconds)
0
50
100
150
200
250
333
3.5 sec
366
2.5 sec
32.1 sec
32.8 sec
451.3 sec
2934 sec
3.2 sec
333
2.3 sec
34.8 sec
30.9 sec
428.2 sec
2720 sec
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
Problem Number

Figure 4

TABLE 3 - COMPARISON OF SOLUTION TIMES

Problem #	Solution time* for I-E w/out S-T(1)	Solution time* for I-E w/S-T(2)	Percent Decrease in Time from (1) to (2)
1	200	184	8
2	217	167	23
3	217	150	30.8
4	284	183	35.2
5	83	66	20.5
6	100	67	33
7	116	83	28.4
8	184	134	27.1
9	333	100	70
10	3.5 sec	3.2 sec	8.6
11	366	333	9
12	2.5 sec	2.3 sec	8
13	34.8 sec	32.1 sec	7.7
14	32.8 sec	30.9 sec	5.7
15	457.1 sec	428.2 sec	5.3
16	2934 sec	2720.0 sec	7.3

* in milliseconds unless otherwise specified

As previously stated, the average percent improvement of I-E w/S-T over I-E w/out S-T was 20.4 %; with a standard deviation of 15%. The effect of using a "good" initial solution for the implicit enumeration routine seems to diminish for the larger problems. That is, for the smaller problems (say, 1-10), the percent improvement is (on the average) much greater than for the other six problems. The least-significant improvement occurred for a 28-variable, 10-constraint problem having 94.3% density. However, the least-significant improvement could amount to considerable savings in computer costs - essentially 5%.

It should be noted that a normal distribution is used for all statistical computations.

CONCLUSIONS

Unless optimal solutions are mandatory, implicit enumeration may not necessarily provide more "beneficial" results than Senju-Toyoda. One should remember that implicit enumeration requires much more time, and thus incurs additional costs.

It is necessary to decide which is the most important factor: an optimal solution or economy of obtaining a solution. Before this is done, one must consider the accuracy of the data. It is highly probably that unless the data is very accurate and precise, that an "optimal" solution does not have a great deal of significance. In this case, why spend money on implicit enumeration?

In most cases presented in this paper, Senju-Toyoda provided an optimal solution. The non-optimal solutions were very close to optimal. Again, this suggests that implicit enumeration may not be the best economical technique.

The results from implicit enumeration, when initialized with the solution from Senju-Toyoda, always required less computation time than implicit enumeration without any good starting solution. This fact seems independent of the problem size and/or the density. The percent improvement with the solution from Senju-Toyoda was greatest for problems with 10 or fewer variables and 5 or fewer constraints. The larger problems did not result in as significant improvements (recall that the minimum improvement was 5.3%).

If finding an optimum solution (if one exists) is the primary importance, initializing implicit enumeration with the solution obtained from Senju-Toyoda will require at least 5% less computer time than implicit enumeration considered separately. Recall that the average improvement was 20.4%. Therefore, if optimality must be guaranteed (again, if there is an optimum), it is suggested that the heuristic method of Senju-Toyoda be used to provide an initial solution to the implicit enumeration routine. However, if "near-optimum" results are sufficient, then only Senju-Toyoda should be used.

APPENDIX A

A Capital Budgeting Problem-Formulation

A firm is considering manufacturing five different types of electronic equipment in its next fiscal year. Resource limitations do not permit all types of equipment to be developed and manufactured. The problem is to decide which equipment types should be developed. The only limitations considered are: (1) engineering manhours, (2) production-line manhours, (3) R & D cost, and (4) materials cost. The resource limitations are:

(1) 4000 engineering manhours

(2) 10,000 production-line manhours

(3) \$180,000--R & D cost

(4) \$100,000--materials cost.

Let $X_j = 1$ if activity j is undertaken and 0 otherwise. The profit from each project, if performed, is given in the objective function. Also, the amount of each resource depleted for performing each activity is shown in the constraints below.

The problem formulation is:

$$\text{Maximize } Z = \sum (\text{Profit})\, jX_j = 80{,}000\, X_1 + 100{,}000\, X_2 + 30{,}000\, X_3 + 60{,}000\, X_4 + 110{,}000\, X_5$$

Subject to:

$$\begin{array}{lrcl}
\text{(Eng. man-hr)} & 800\, X_1 + 1{,}500\, X_2 + 400\, X_3 + 700\, X_4 + 1{,}200\, X_5 & \leq & 4{,}000 \\
\text{(Prod-line hrs)} & 1500\, X_1 + 3{,}000\, X_2 + 500\, X_3 + 3{,}000\, X_4 + 4{,}000\, X_5 & \leq & 10{,}000 \\
\text{(R \& D cost)} & 30{,}000\, X_1 + 60{,}000\, X_2 + 10{,}000\, X_3 + 40{,}000\, X_4 + 75{,}000\, X_5 & \leq & 180{,}000 \\
\text{(Mat. cost)} & 25{,}000\, X_1 + 30{,}000\, X_2 + 15{,}000\, X_3 + 20{,}000\, X_4 + 30{,}000\, X_5 & \leq & 100{,}000
\end{array}$$

$$X_j = 0 \text{ or } 1.$$

APPENDIX B

Example Using Implicit Enumeration

Let the problem be formulated/defined as

$$\begin{array}{ll} \text{Minimize} & 5X_1 + 7X_2 + 10X_3 + 3X_4 + X_5 = f \\ \text{S. T.} & X_1 - 3X_2 + 5X_3 + X_4 - 4X_5 \geq 2 \\ & -2X_1 + 6X_2 - 3X_3 - 2X_4 + 2X_5 \geq 0 \\ & -X_2 + 2X_3 - X_4 - X_5 \geq 1 \\ & X_j = 0,1 \end{array}$$

The following steps correspond to the ones in the detailed flow diagram presented earlier.

<u>STEP</u>

(1) XBAR = (1,1,1,1,1)   ZBAR = 26 = f(1,1,1,1,1) S = $\emptyset$

(2) Constraints 1 and 3 are violated when all variables equal 0;

V = (1,3)

(3) V $\neq \emptyset$

(4) f_p = 0 since all variables equal 0

limit = ZBAR - f_p = 26

(5) i) (1,2,3,4,5)

ii) (1,3,4) ---T = (1,3,4)

(6) T $\neq \emptyset$

(7) Yes, since 1 + 5 + 1 - 2 = 5 0 --OK

2 - 1 - 1 = 0 0 --OK

(8) Choose variable with the greatest coefficient sum

X_1: 1 - 2 + 0 = -1

X_3: 5 - 3 + 2 = 4* ---S = (3)

X_4: 1 - 2 - 1 = -2

(2) $X_3 = 1$; $X_1 = X_2 = X_4 = X_5 = 0$ $V = (2)$

(3) $V \neq \emptyset$

(4) $f_p = 10$ limit = ZBAR - f_p = 26 - 10 = 16

(5) i) (1,2,4,5)

ii) (2,5) --- T = (2,5)

(6) $T \neq \emptyset$

(7) Constraint 2 can be made feasible by adding to S only variables in T

(8) S = (3,2)

(2) $V = \emptyset$

(3) Yes

(9) XBAR = (0,1,1,0,0) ZBAR = 7 + 10 = 17

(10) S = (3,-2) -2 = complement of 2, which means $X_2 = 0$

(2) V = (2) (because $X_3 = 1$; $X_1 = X_2 = X_4 = X_5 = 0$)

(3) $V \neq \emptyset$

(4) $f_p = 10$ limit = ZBAR - f_p = 17 - 10 = 7

(5) i) (1,4,5)

ii) (5) --- T = (5)

(6) $T \neq \emptyset$

(7) The second constraint is infeasible when $X_5 = 1$ is added to S:

$-3 + 2 \not\geq 0$

(11) No

(10) S = (-3) which implies $X_3 = 0$. Note that all elements to the right of -3 are dropped.

(2) V = (1,3)

(3) $V \neq \emptyset$

(4) $f_p = 0$ limit ZBAR - f_p = 17

(5) i) (1,2,3,4,5)

ii) (1,4) --- T (1,4)

(6) $T \neq \emptyset$

(7) No

(11) Yes

(12) Stop. The solution is optimal.

XBAR = (0,1,1,0,0) ---$X_1 = X_4 = X_5 = 0$; $X_2 = X_3 = 1$

ZBAR = 17

APPENDIX C1

The Assignment Problem

The assignment problem can be formulated as:

Maximize or Minimize $f = \sum_{i=1}^{N} \sum_{j=1}^{N} C_{ij}X_{ij}$

Subject to:

$$\sum_{i=1}^{N} X_{ij} = 1 \quad \text{for all } j$$

$$\sum_{j=1}^{N} X_{ij} = 1 \quad \text{for all } i$$

$$X_{ij} = 0,1 \text{ for all } i,j$$

where n = number of assignments.

Example: Suppose there are four employees to be assigned to four tasks so that each employee receives exactly one task. The objective is to minimize the total cost, where a known cost exists for each possible assignment. The following table displays the costs:

Costs in Dollars

Employee	Tasks	1	2	3	4
Bob		5	3	7	8
Alice		4	4	6	9
Ted		7	8	10	7
Carol		5	5	12	8

Formulate the problem in terms of double subscripts where the first subscript refers to the row or employee, and the second subscript refers to the column or task. Also, define X_{ij} to be the number of times (0 or 1) that employee i is assigned to task j. Then the objective function is

$$\text{Minimize } f = 5x_{11} + 3x_{12} + 7x_{13} + 8x_{14}$$
$$+ 4x_{21} + 4x_{22} + 6x_{23} + 9x_{24}$$
$$+ 7x_{31} + 8x_{32} + 10x_{33} + 7x_{34}$$
$$+ 5x_{41} + 5x_{42} + 12x_{43} + 8x_{44}$$

The first constraint type relates to the requirement that each employee shall receive a task:

Bob: $x_{11} + x_{12} + x_{13} + x_{14} = 1$

Alice: $x_{21} + x_{22} + x_{23} + x_{24} = 1$

Ted: $x_{31} + x_{32} + x_{33} + x_{34} = 1$

Carol: $x_{41} + x_{42} + x_{43} + x_{44} = 1$

The following constraints allow only one task to be assigned to each employee:

Task 1: $x_{11} + x_{21} + x_{31} + x_{41} = 1$

Task 2: $x_{12} + x_{22} + x_{32} + x_{42} = 1$

Task 3: $x_{13} + x_{23} + x_{33} + x_{43} = 1$

Task 4: $x_{14} + x_{24} + x_{34} + x_{44} = 1$

$$x_{ij} = 0 \text{ or } 1 \text{ for all } i,j$$

APPENDIX C2

The Knapsack Problem

A salesman wishes to pack his carrying case - his knapsack - so that he gets as much value into the case as possible. He does not want to carry more than 25 pounds of merchandise and he can take no more than one of each item. The objective is to pack his case in the "best" manner. The following table provides the value and weight of the possible items the salesman can carry.

ITEM #	VALUE ($)	WEIGHT (LBS.)
1	$ 35	1
2	85	4
3	117	17
4	24	2
5	94	3
6	10	4

The problem is formulated as:

Maximize $35x_1 + 85x_2 + 117x_3 + 24x_4 + 94x_5 + 10x_6$

Subject to $1x_1 + 4x_2 + 17x_3 + 2x_4 + 3x_5 + 4x_6 \leq 25$

$x_j = 0 \text{ or } 1.$

Note that additional constraints could include volume, width, or bulkiness.

APPENDIX C3

The Traveling Salesman Problem

Suppose a salesman must visit several cities in his sales circuit. He is interested in knowing how to sequence his visits to all of the cities such that each city is visited exactly once and the total travel distance is minimized (and thus minimizing travel cost). When this problem is formulated as a 0-1 integer problem, the number of constraints grows very rapidly as the number of cities increases. The example below deals with a three-city problem and the distances are in the table below:

DISTANCES

		TO CITY 1	2	3
FROM CITY	1	-	25	40
	2	25	-	10
	3	30	10	-

$$\text{Let } X_{ij} = \begin{cases} 1 \text{ if the salesman travels from i to j} \\ 0 \text{ otherwise} \end{cases}$$

The objective function is to minimize the total distance (and thus total cost).

Minimize $f = 25x_{12} + 40x_{13} + 25x_{21} + 10x_{23} + 30x_{31} + 10x_{32}$

The first constraint type requires that each city is left once. (each row sum equals 1):

$$x_{12} + x_{13} = 1$$
$$x_{21} + x_{23} = 1$$
$$x_{31} + x_{32} = 1$$

The second type of constraint requires that each city is visited exactly once (column sum equals 1).

$$x_{21} + x_{31} = 1$$
$$x_{12} + x_{32} = 1$$
$$x_{13} + x_{23} = 1$$

To eliminate subtours we need:

$$x_{12} + x_{21} \leq 1$$
$$x_{13} + x_{31} \leq 1$$
$$\cdot$$
$$x_{23} + x_{32} \leq 1$$
$$x_{ij} = 0,\ 1 \text{ for all } i,j$$

APPENDIX D

A listing of the FORTRAN IV main program that calls the implicit enumeration and Senju-Toyoda routines is included in the following pages. The method of Senju-Toyoda is also included.

The main steps of the calling program are:

1. Initialize and input data (in Senju-Toyoda format)
2. Call SENJU--which in turn calls subroutine SORT SORT orders the vectors from SENJU.
3. Compute the time for Senju-Toyoda; find sum of objective function coefficients for ZBAR used in implicit enumeration; output data
4. Convert to Balas format
5. Call BALL with ZBAR as bound
6. Compute time for Balas; output data
7. Take solution vector and ZBAR to subroutine BALL; call BALL with these known values
8. Compute time; output data

LITERATURE CITED

1. Senju, S., and Toyoda, Y., " An Approach to Linear Programming with 0-1 Variables ": Management Science, V. 15, No. 4, p. B-196 to B-208 (1968)

2. Balas, E., "An Additive Algorithm for Solving Linear Programs with Zero-One Variables": Operations Research, V. 13, No. 3, pp. 517-546 (1965)

3. Hadley, G. Nonlinear and Dynamic Programming, Addison-Wesley Publishing Co., p. 292 (1964)

4. Balinsky, M. L., "Integer Programming: Methods, Uses, Computations:" Management Science, V. 12, No. 3, pp. 253-313

5. Ford, L. R. Jr., and Fulkerson, D. R., Flows in Networks, Princeton University Press, (1962)

6. Geoffrion, A. M., "An Improved Implicit Enumeration Approach for Integer Programming: Operations Research, V. 17, No. 3, pp. 437-454 (1969)

7. Glover, F., An Multiphase-Dual Algorithm for the Zero-One Integer Programming Problem ": Operations Research, V. 13, No. 6. pp. 879-919 (1965)

8. Lawler, E. L., and Wood, D. E., "Branch-and-Bound Methods: A Survey " Operations Research, V. 14, No. 4, pp. 699-719 (1966)

9. Peterson, C. C., " Computational Experience with Variants of the Balas Algorithm Applied to the Selection of R and D Projects": Management Science, V. 13, No. 6, pp. 736-750 (1967)

10. Davis, R. E., Kendrick, D. A., and Weitzman, M. A., " Branch-and-Bound Algorithm for Zero-One Mixed Integer Programming Problems " Operations Research, V. 19, No. 4, pp. 1036-1044 (1971)

11. Everett, H., III, " General LaGrange Multiplier Method for Solving Problems of Optimum Allocation of Resources" Operations Research, V. 11, No. 3, pp. 399-417 (1963)

12. Wiley. R. W., "Searching for the Multipliers in Linear Binary-Integer Problems with Linear Programming": Master's Thesis, Colorado School of Mines, Golden, Colorado (1972)

13. Hiller, F. S., and Lieberman, G. J., Introduction to Operations Research, Holden-Day, Inc., (1970)

14. Trauth, C. A., Jr., and Woolsey, R. E. D., "Integer Linear Programming A Study in Computational Efficiency" Management Science, V. 15, p. 481, (1968)

15. Haldi, J., "Twenty-five Integer Programming Test Problems": Working Paper No. 43, Graduate School of Business, Stanford University (1964)

```
      INTEGER V,V1
      DIMENSION IXB(70),IX(50),LFORM(20)
      COMMON/SENJU1/A(50,50),B(50),C(50)
C
      V=7
      V1=6
C
66    IF(V .EQ. 8) GO TO 32
1     FORMAT(2I)
2     FORMAT(50G)
3     FORMAT(10F8.1)
5     FORMAT(18X,'X(',I3,')=',2X,I2)
6     FORMAT(//5X,'THE RESULTS FROM SENJU-TOYODA ARE',//)
7     FORMAT(//,' THE VALUE OF THE OBJECTIVE FUNCTION IS',F11.3)
8     FORMAT('  ENTER "1" IF A(I,J)/B(I) IS DESIRED .',/)
9     FORMAT(/' THE TIME FOR SENJU-TOYODA WAS ',I11,' MILLISECONDS'//)
40    FORMAT('  WHAT ARE THE VALUES FOR M AND N ??'/)
41    FORMAT('  INPUT A(I,J) AND B(I) .',/)
42    FORMAT('  INPUT C(J) .',/)
43    FORMAT(1X,30('\/'))
44    FORMAT(1H1)
45    FORMAT(/)
46    FORMAT(20A5)
47    FORMAT('  ENTER INPUT FORMAT.',/)
      IF(V .EQ. 4) WRITE(V1,8)
      READ(V,1) K1
      IF(V .EQ. 4) WRITE(V1,40)
      READ(V,1) M,N
      IF(V .EQ. 4) WRITE(V1,47)
      READ(V,46) (LFORM(I),I=1,20)
      IF(V .EQ. 4) WRITE(V1,41)
      DO 10 I=1,M
      READ(V,LFORM) (A(I,J),J=1,N),B(I)
10    CONTINUE
      IF(V .EQ. 4) WRITE(V1,42)
      READ(V,LFORM) (C(J),J=1,N)
      IF(K1 - 1) 11,12,11
12    DO 13 I=1,M
      IF(B(I) .EQ. 0.0) GO TO 13
      DO 13 J=1,N
      A(I,J)=A(I,J)/B(I)
13    CONTINUE
11    CONTINUE
      WRITE(V1,3) (C(J),J=1,N)
      WRITE(V1,45)
      DO 16 I=1,M
      IF(K1 .EQ. 1) B(I)=1
      WRITE(V1,3) (A(I,J),J=1,N),B(I)
16    WRITE(V1,45)
      CALL RTIME (IT1)
C
```

```
C
         CALL SENJU  (M,N,IXB)
C
C
         CALL RTIME (IT2)
         ITS=IT2-IT1
         Z=Ø.Ø
         DO 25 J=1,N
25       Z=Z+C(J)
         ZBAR=Z
         ZBAR1=Z
         ABC=Ø.Ø
         DO 24 I=1,M
         DO 23 J=1,N
23       ABC=ABC+A(I,J)
         B(I)=B(I)-ABC
         ABC=Ø.Ø
24       CONTINUE
         DD=Ø.Ø
C
C
         CALL RTIME(IT3)
C
         CALL BALL (A,B,C,IX,ZBAR,M,N,DD)
C
         CALL RTIME (IT4)
         ITS1=IT4-IT3
C
C
C        DO 5Ø I=1,M
C5Ø      WRITE(V1,3) (A(I,J),J=1,N),B(I)
C
C
131      OB=Ø.Ø
         DO 14 J=1,N
         EIX=IXB(J)
14       OB=OB+C(J)*EIX
         WRITE(V1,6)
         DO 15 J=1,N
15       WRITE(V1,5) J,IXB(J)
         WRITE(V1,7) OB
         WRITE(V1,9) ITS
         WRITE(V1,43)
         WRITE(V1,2Ø)
         DO 22 J=1,N
         IX(J)=1-IX(J)
22       WRITE(V1,5) J,IX(J)
         Z=Z-ZBAR
         WRITE(V1,7) Z
         IF(ITS1-1ØØØ) 34,34,35
34       WRITE(V1,27) ITS1
```

```
      GO TO 36
35    AITS1=ITS1
      AITS1=AITS1/1000.
      WRITE(V1,28) AITS1
36    CONTINUE
C
C
      ZBAR=OB
      Z=OB
C
C
      DO 26 J=1,N
26    IX(J)=1-IXB(J)
      DD=1.0
      CALL RTIME (IT5)
      CALL BALL (A,B,C,IX,ZBAR,M,N,DD)
      CALL RTIME (IT6)
      ITS1=IT6-IT5
C
      WRITE(V1,43)
      WRITE(V1,30)
      RBC=IX(1)
      DO 29 J=1,N
29    IF(IX(J) .EQ. 9) IX(J)=1-IXB(J)
      DO 31 J=1,N
      IX(J)=1-IX(J)
31    WRITE(V1,5) J,IX(J)
      ZBAR=0.0
      DO 350 J=1,N
      EIX=IX(J)
350   ZBAR=ZBAR+C(J)*EIX
C     ZBAR=ZBAR1-ZBAR
      IF(RBC .EQ. 9.0) ZBAR=OB
      WRITE(V1,7) ZBAR
      IF(ITS1 - 1000) 37,37,38
37    WRITE(V1,32) ITS1
      GO TO 39
38    AITS1=ITS1
      AITS1=AITS1/1000.
      WRITE(V1,33) AITS1
39    CONTINUE
      IF(V1 .EQ. 6) WRITE(V1,44)
      V=V+1
      GO TO 66
20    FORMAT(//,1X,'THE RESULTS FROM IMPLICIT ENUMERATION W/OUT
     1 SENJU-TOYODA ARE',//)
27    FORMAT(/,' THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS',I5,
     2 ' MILLISECONDS',//)
28   FORMAT(/,' THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS',F5.1,
     1 ' SECONDS',//)
30    FORMAT(//,1X,'THE RESULTS FROM IMPLICIT ENUMERATION WITH
```

```
     2 SENJU-TOYODA ARE',//)
32      FORMAT(/,' THE TIME FOR BALAS WITH SENJU-TOYODA WAS',I6,
     2  ' MILLISECONDS',////)
33     FORMAT(/,' THE TIME FOR BALAS WITH SENJU-TOYODA WAS ',F5.1,
     2 ' SECONDS',////)
        RETURN
        END
```

```
      SUBROUTINE SENJU (M,N,IXB)
      DIMENSION IXB(70),R(50),S(50),G(50),AI(50)
      COMMON/SENJU1/A(50,50),B(50),C(50)
      ERR=0.000001
      DO 4 J=1,N
4     IXB(J)=1
C
C
C     SET UP THE VECTORS
C
C
      S1=0.0
      DO 8 I=1,M
      SS=0.0
      DO 5 J=1,N
5     SS=SS+A(I,J)
      R(I)=SS
      TT=SS-B(I)
      IF(TT-ERR) 6,6,7
6     TT=0.0
      GO TO 8
7     S1=S1+TT*TT
8     S(I)=TT
      IF(S1-ERR) 30,30,9
9     S1=1./SQRT(S1)
C     DO 10 I=1,M
C10   S(I)=S(I)*S1
C     FIND U-VECTOR AND EFFECTIVE GRADIENTS
C
      DO 12 J=1,N
      U=0.0
      DO 11 I=1,M
11    U=U+A(I,J)*S(I)
      G(J)=C(J)/U
12    AI(J)=J
C
C
C     ORDER THE VECTORS
C
C
      CALL SORT (G,AI,N)
      IN=1
C
C
C     DETERMINE WHICH VARIABLES ARE TO BE SET TO ZERO
C
C
17    JJ=AI(IN)
      IND=1
      DO 14 I=1,M
      R(I)=R(I)-A(I,JJ)
```

```
      IF(R(I)-B(I)) 14,14,13
13    IND=2
14    CONTINUE
      IXB(JJ)=0
      IF(IND .EQ. 1) GO TO 15
16    IN=IN+1
      GO TO 17
15    DO 18 I=1,M
18    S(I)=R(I)-B(I)
      JN=IN-1
      DO 25 K=1,JN
      IN=IN-1
      JJ=AI(IN)
      DO 20 I=1,M
      IF(S(I)+A(I,JJ)) 20,20,25
20    CONTINUE
      IXB(JJ)=1
      DO 21 I=1,M
21    S(I)=S(I)+A(I,JJ)
25    CONTINUE
30    RETURN
      END
C
C
C
```

```
      SUBROUTINE SORT (AG,B,N)
C
C
C     THIS SUBROUTINE ORDERS ANY VECTOR AG WITH N ELEMENTS AND PUTS
C     THE ELEMENTS OF B IN THE CORRESPONDING ORDER.
C
C
      DIMENSION AG(50),B(50)
      NN=N-1
      DO 4 I=1,NN
      AMIN=AG(I)
      IND=1
      II=I+1
      DO 2 J=II,N
      IF(AMIN-AG(J)) 2,2,1
1     AMIN=AG(J)
      IHOLD=J
      IND=2
2     CONTINUE
      GO TO (4,3) IND
3     BMIN=B(IHOLD)
      AG(IHOLD)=AG(I)
      B(IHOLD)=B(I)
      AG(I)=AMIN
      B(I)=BMIN
4     CONTINUE
      RETURN
      END
```

```
      SUBROUTINE BALL (A,B,C,IX,ZBAR,M,N,DD)
      INTRVL=1
      INTEGER V,V1
      DIMENSION A(50,50),C(50),B(50),CS(50),W(50,50),IX(50),IV(50)
      DIMENSION IT(50),NOTT(50),SUMS(50),IS(50)
      DIMENSION IPRINT(50),ISAVE(50,50),ISTEP(50),INUM(50)
      V=4
      V1=4
C
      EPS=0.000001
C
C
      DO 11 I=1,50
11    INUM(I)=I
1     CONTINUE
      ITPCK=0
      IFEAS=0
      ICOUNT=0
C     IF(V1 .EQ. 4) WRITE(V1,501)
501   FORMAT(///,'  INPUT M,N,INTERVAL,',/)
C     READ(V,500) M,N,INTRVL
500   FORMAT(3I)
C     IF(M-51) 4,9000,9000
4     DO 2 I=1,50
C     B(I)=0.0
C     C(I)=0.0
      IS(I)=0
      IV(I)=0
      IT(I)=0
      IF(DD .EQ. 1) GO TO 8
      IX(I)=9
8     NOTT(I)=0
      SUMS(I)=0.0
      DO 2 J=1,50
C     A(I,J)=0.0
      W(I,J)=0.0
2     CONTINUE
C
      DO 3 I=34
      IPRINT(I)=0
3     CONTINUE
C     IF(V .EQ. 4) WRITE(V1,511)
511   FORMAT(/,'  INPUT C(J),',/)
C     READ(V,510) (C(J),J=1,N)
C     IF(V .EQ.4) WRITE(V1,502)
502   FORMAT(/,'  INPUT B(I),A(I,J).',/)
C     DO 10 I=1,M
C10   READ(V,510) B(I),(A(I,J),J=1,N)
C     IF(V .EQ. 4) WRITE(V1,512)
512   FORMAT(/,'  INPUT ZBAR.',/)
C     READ(V,510) ZBAR
```

```
510     FORMAT(50G)
        FZBAR=ZBAR
        DO 20 J=1,N
        CS(J)=0.0
        DO 20 I=1,M
20      CS(J)=CS(J)+A(I,J)
C
C
C       THIS SECTION PRINTS MATRIX INPUT
C
C
C
C       IF(V1 .EQ. 4) WRITE(V1,12)
C       IF(V1 .EQ. 4) WRITE(V1,13) (INUM(J),J=1,N)
C       IF(V1 .EQ. 4) WRITE(V1,14) (C(J),J=1,N)
C       IF(V1 .EQ. 4) WRITE(V1,15)
12      FORMAT(///2X,18HOBJECTIVE FUNCTION,/)
13      FORMAT(2X,10(4X,1HX,I2))
14      FORMAT(/,2X,10F6.1,/)
15      FORMAT(/,2X,11HCONSTRAINTS,//,6X,8HCONSTANT,/)
C
C
C       DO 84 I=1,M
C       IF(V1 .EQ. 4) WRITE(V1,16) I,B(I),(A(I,J),J=1,N)
84      CONTINUE
16      FORMAT(1X,1HG,I2,2X,F6.1,10F8.1,/)
C
C
        IF(DD .EQ. 0) GO TO 36
        DO 35 J=1,N
        IF(IX(J) .EQ. 1) IS(J)=J
35      IF(IX(J) .EQ. 0) IS(J)=0
36      CONTINUE
        DO 17 I=1,M
        IF(B(I)) 19,17,17
17      CONTINUE
        DO 18 I=1,50
        IX(I)=0
18      CONTINUE
        ZBAR=0.0
C       WRITE(V1,22)
22      FORMAT(//,10X,26HALL CONSTANTS ARE POSITIVE,/)
        GO TO 1750
19      IF(V1 .EQ. 6) WRITE(V1,21)
21      FORMAT(/,6H STEP*,34X,1H*,34X,9H*OBJ FCN*,34X,11H* NOT* ADD*,
     2  /,6H NUMB*,6X,22HPARTIAL SOLUTION   (S),6X,1H*,4X,26HVIOLATED CO
     3  TRAINTS   (V),4X,9H*COF LIM*,7X,20HVARIABLES IN SET (T),
     4  7X,17H* SAT*TO S*  ZBAR,/,135(1H*))
C
C
        NUMB=0
```

```
      NS=0
C
C     STEP 2
C
C     FIND V, THE SET OF CONSTRAINTS VIOLATED WHEN PARTIAL
C     SOLUTION S IS COMPLETED BY SETTING TO ZERO ALL VARIABLES
C     NOT IN THE SET S.
C
C     FIND FP, THE VALUE OF F WHEN S IS COMPLETED BY SETTING
C     TO ZERO ALL VARIABLES NOT IN S.
C
C
45    IF(NUMB) 645,645,639
639   IP=11
      IF(NS-11) 640,640,642
640   IP=NS
642   DO 1001 I=1,IP
      IPRINT(I)=IS(I)
1001  CONTINUE
645   FP=0.0
      NW=0
      IF(NS) 51,51,52
52    DO 50 J=1,NS
      IF(IS(J)) 50,50,55
55    NW=NW+1
      JJ=IS(J)
      DO 60 I=1,M
60    W(I,NW)=A(I,JJ)
      FP=FP+C(JJ)
50    CONTINUE
51    NW=NW+1
      DO 65 I=1,M
65    W(I,NW)=B(I)
      MV=0
      DO 70 I=1,M
      SUMS(I)=0.0
      DO 80 J=1,NW
80    SUMS(I)=SUMS(I)+W(I,J)
      IF(SUMS(I)+EPS) 85,70,70
85    MV=MV+1
      IV(MV)=I
70    CONTINUE
C
C     STEP 3
C
C     IS THE SET V EMPTY ?
C     IF YES--GO TO STEP 9
C     IF NO --GO TO STEP 4
C
      IF(MV) 200,200,90
90    IP=11
```

```
        IF(MV-11) 92,92,94
92      IP=MV
94      DO 1200 I=1,IP
        IPRINT(I+11)=IV(I)
1200    CONTINUE
C
C       STEP 4
C
C       SET THE OBJECTIVE FUNCTION COEFFICIENT LIMIT TO ZBAR-FP,
C
        CLIM=ZBAR-FP
        NW=0
        NT=0
        IT(1)=0
C
C       STEP 5
C
C       STORE IN THE SET T EACH VBL NOT IN THE SET S WHICH HAS
C               1. AN OBJ FCN COEFF. LESS THAN THE LIMIT
C               2. A POSITIVE COEFF. IN SOME CONSTRAINT V
C
        DO 100 J=1,N
100     NOTT(J)=0
        IF(NS) 104,104,101
101     DO 105 J=1,NS
        ITEMP=IS(J)
        IF(ITEMP) 102,105,105
102     ITEMP=-ITEMP
105     NOTT(ITEMP)=1
C       IF C(J) IS LESS THAN OR EQUAL TO CLIM: 110,120,120
104     DO 110 J=1,N
        IF(NOTT(J)) 115,115,110
115     IF(CLIM-C(J)) 110,120,120
120     DO 125 I=1,MV
        ITEMP=IV(I)
        IF(A(ITEMP,J)) 125,125,130
125     CONTINUE
        GO TO 110
130     NT=NT+1
        IT(NT)=J
        NW=NW+1
        DO 135 I=1,M
135     W(I,NW)=A(I,J)
110     CONTINUE
        IP=11
        IF(NT-11) 106,106,108
106     IP=NT
108     DO 1300 I=1,IP
        IPRINT(I+22)=IT(I)
1300    CONTINUE
C
```

```
C         STEP 6
C
C         IS THE SET T EMPTY ?
C         IF YES--SET ITPCK TO 1 AND GO OUTPUT SECTION THEN
C                 GO TO STEP 11 (BACKTRACK)
C         IF NO --GO TO STEP 7
C
          IF(NT) 1400,1400,138
1400      ITPCK=1
          JMAX=0
          GO TO 1000
C
C         STEP 7
C
C         CAN EVERY CONSTRAINT IN V BE MADE FEASIBLE BY ADDING
C         ONLY VBLS IN T ?
C
C         IF NO -SET ITPCK TO 1 AND GO TO OUTPUT SECTION,THEN
C                 GO TO STEP 11
C         IF YES--GO TO STEP 8
C
138       DO 140 I=1,MV
          ITEMP=IV(I)
          DO 145 J=1,NW
          IF(W(ITEMP,J)) 145,145,150
150       SUMS(ITEMP)=SUMS(ITEMP)+W(ITEMP,J)
145       CONTINUE
          IF(SUMS(ITEMP)+EPS) 152,140,140
152       IPRINT(34)=ITEMP
          ITPCK=1
          JMAX=0
          GO TO 1000
140       CONTINUE
C
C         STEP 8
C
C         ADD TO S THE VBL IN T WITH THE GREATEST COEFF, SUM,
C         GO TO OUTPUT SECTION,THEN GO TO STEP 2
C
          JMAX=IT(1)
          CSMAX=CS(JMAX)
          IF(NT-2) 156,146,146
146       DO 155 J=2,NT
          JTEMP=IT(J)
          IF(CS(JTEMP)-CSMAX) 155,160,170
160       IF(C(JTEMP)-C(JMAX)) 170,155,155
170       JMAX=JTEMP
          CSMAX=CS(JTEMP)
155       CONTINUE
156       CONTINUE
          GO TO 1000
```

```
157     CONTINUE
        NS=NS+1
        IS(NS)=JMAX
        NUMB=NUMB+1
        GO TO 45
C
C       STEP 9
C
C       COMPLETE THE PARTIAL SOLUTION S BY SETTING TO ZERO ALL
C       VBLS NOT IN S.  THIS COMPLETED SOLUTION BECOMES THE
C       INCUMBENT SOLUTION X-BAR, AND THE VALUE OF THE OBJ
C       FCN AT X-BAR BECOMES THE NEW VALUE OF ZBAR.
C
200     DO 210 J=1,N
210     IX(J)=0
        ZBAR=0.0
        DO 215 J=1,NS
        JTEMP=IS(J)
        IF(JTEMP) 215,215,217
217     IX(JTEMP)=1
        ZBAR=ZBAR+C(JTEMP)
215     CONTINUE
C
C       FEASIBLE SOLUTION ENCOUNTERED--SET IFEAS TO 1 TO SAVE
C
218     IFEAS=1
        JMAX=0
        CLIM=0.0
        DD=0.0
C
C       THIS SECTION IS THE OUTPUT SECTION
C
        INT=INTRVL
1000    ICK=(NUMB/INT)*INT-NUMB
        IF(ICK) 1550,1010,1550
1010    IF(V1 .EQ.6) WRITE(V1,1500) NUMB,(IPRINT(I),I=1,11),(IPRINT(J),
      2 J=12,22),CLIM,(IPRINT(K),K=23,33),IPRINT(34),JMAX,ZBAR
1500    FORMAT(1X,I3,2H *,2(11I3,2H *),F6.1,2H *,11I3,2H *,
      2 2(I3,2H *),F6.1,/,5X,1H*,2(34X,1H*),7X,1H*,34X,1H*,2(4X,1H*))
1550    DO 1600 I=1,34
        IPRINT(I)=0
1600    CONTINUE
        IF(IFEAS-1) 1605,300,300
1605    IF(ITPCK-1) 157,300,300
C
C       STEP 11
C
C       ARE ALL ELEMENTS IN THE SET S NEGATIVE?
C
C       IF NOT--LOCATE THE RIGHTMOST POS. ELEMENT IN S.
C       REPLACE IT WITH ITS COMPLEMENT AND DROP ANY
```

```
C        ELEMENTS TO THE RIGHT.  THEN GO TO STEP 2.
C
C        IF SO--TERMINATE.
C
300      NEWS=NS
         DO 220 J=1,NS
         JJ=NS-J+1
         IF(IS(JJ)) 225,225,230
225      NEWS=NEWS-1
220      CONTINUE
         GO TO 400
230      IS(JJ)=-IS(JJ)
         NS=NEWS
         IF(IFEAS-1) 1512,1508,1508
1508     IF(ITPCK-1) 1511,1512,1512
1511     IF(50-ICOUNT) 1512,1512,1509
1509     ICOUNT=ICOUNT+1
         ISTEP(ICOUNT)=NUMB
         DO 1510 I=1,N
         ISAVE(ICOUNT,I)=IX(I)
1510     CONTINUE
1512     IFEAS=0
         ITPCK=0
         NUMB=NUMB+1
         GO TO 45
C
C        STEP 12
C
C        TERMINATE--THE INCUMBENT SOLUTION, IF ANY, IS OPTIMAL.
C        IF NONE--THEN THERE IS NO FEASIBLE SOLUTION BETTER THAN
C                 THE INITIAL VALUE OF ZBAR.
C
400      IF(V1 .EQ. 6) WRITE(V1,1610)
1610     FORMAT(1H0)
         IF(IX(1)-9) 1630,1615,1615
1615     WRITE(V1,1620) FZBAR
1620     FORMAT(5X,'THERE IS NO FEASIBLE SOLUTION WITH A VALUE FOR
     2 THE OBJECTIVE FUNCTION LOWER THAN',F7.1,', THE INITIAL
     3 ZBAR VALUE.')
         GO TO 9000
         GO TO 1
1630     DO 1700 I=1,ICOUNT
C        IF(V1.EQ.6) WRITE(V1,1650) ISTEP(I),(ISAVE(I,J),J=1,N)
1650     FORMAT(//,4X,'FEASIBLE SOLUTION, STEP',I4,2X,50I2)
1700     CONTINUE
C        WRITE(V1,1770)
1750     DO 1760 I=1,N
C        WRITE(V1,1800) I,IX(I)
1760     CONTINUE
1800     FORMAT(18X,'X(',I3,')=',I4)
1770     FORMAT(//,10X,'THE OPTIMAL SOLUTION IS')
```

```
C       WRITE(V1,1900) ZBAR
1900    FORMAT(//10X,'OPTIMAL VALUE OF OBJECTIVE FUNCTION =',F10.4)
C       GO TO 1
9000    RETURN
        END
```

```
20.0     18.0     17.0     15.0     15.0     10.0     5.0     3.0     1.0

30.0     25.0     20.0     18.0     17.0     11.0     5.0     2.0     1.0
67.0


 THE RESULTS FROM SENJU-TOYODA ARE


                X(  1)=    0
                X(  2)=    0
                X(  3)=    1
                X(  4)=    0
                X(  5)=    1
                X(  6)=    1
                X(  7)=    1
                X(  8)=    1
                X(  9)=    1
                X( 10)=    1
THE VALUE OF THE OBJECTIVE FUNCTION IS:      52.000

THE TIME FOR SENJU-TOYODA WAS          17 MILLISECONDS


↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/


THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE


                X(  1)=    0
                X(  2)=    0
                X(  3)=    1
                X(  4)=    1
                X(  5)=    1
                X(  6)=    1
                X(  7)=    0
                X(  8)=    0
                X(  9)=    1
```

```
                X( 10)=    0
THE VALUE OF THE OBJECTIVE FUNCTION IS:     58.000

THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS  200 MILLISECONDS


↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/


THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE


                X(  1)=    0
                X(  2)=    0
                X(  3)=    1
                X(  4)=    1
                X(  5)=    1
                X(  6)=    1
                X(  7)=    0
                X(  8)=    0
                X(  9)=    1
                X( 10)=    0
THE VALUE OF THE OBJECTIVE FUNCTION IS:     58.000

THE TIME FOR BALAS WITH SENJU-TOYODA WAS   184 MILLISECONDS
```

20.0	18.0	17.0	15.0	15.0	10.0	5.0	3.0	1.0
30.0	25.0	20.0	18.0	17.0	11.0	5.0	2.0	1.0
80.0								

THE RESULTS FROM SENJU-TOYODA ARE

```
X(  1)=   0
X(  2)=   0
X(  3)=   1
X(  4)=   1
X(  5)=   1
X(  6)=   1
X(  7)=   1
X(  8)=   1
X(  9)=   1
X( 10)=   1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 67.000

THE TIME FOR SENJU-TOYODA WAS   0 MILLISECONDS

↑/

THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE

```
X(  1)=   0
X(  2)=   1
X(  3)=   1
X(  4)=   0
X(  5)=   1
X(  6)=   1
X(  7)=   1
X(  8)=   1
X(  9)=   0
```

```
                 X( 10)=    0
THE VALUE OF THE OBJECTIVE FUNCTION IS:      68.000

THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS  217 MILLISECONDS


↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/


THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE


                 X(  1)=    0
                 X(  2)=    1
                 X(  3)=    1
                 X(  4)=    0
                 X(  5)=    1
                 X(  6)=    1
                 X(  7)=    1
                 X(  8)=    1
                 X(  9)=    0
                 X( 10)=    0
THE VALUE OF THE OBJECTIVE FUNCTION IS:      68.000

THE TIME FOR BALAS WITH SENJU-TOYODA WAS   167 MILLISECONDS
```

```
20.0    18.0    17.0    15.0    15.0    10.0    5.0    3.0    1.0

30.0    25.0    20.0    18.0    17.0    11.0    5.0    2.0    1.0
70.0


 THE RESULTS FROM SENJU-TOYODA ARE

          X(  1)=   0
          X(  2)=   0
          X(  3)=   1
          X(  4)=   0
          X(  5)=   1
          X(  6)=   1
          X(  7)=   1
          X(  8)=   1
          X(  9)=   1
          X( 10)=   1
THE VALUE OF THE OBJECTIVE FUNCTION IS:     52.000

THE TIME FOR SENJU-TOYODA WAS          0 MILLISECONDS


↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/


THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE


          X(  1)=   0
          X(  2)=   0
          X(  3)=   1
          X(  4)=   1
          X(  5)=   1
          X(  6)=   1
          X(  7)=   0
          X(  8)=   1
          X(  9)=   1
```

```
                    X( 10)=    1
THE VALUE OF THE OBJECTIVE FUNCTION IS:     62.000

THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS  217 MILLISECONDS


↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/


THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE


                    X(  1)=    0
                    X(  2)=    0
                    X(  3)=    1
                    X(  4)=    1
                    X(  5)=    1
                    X(  6)=    1
                    X(  7)=    0
                    X(  8)=    1
                    X(  9)=    1
                    X( 10)=    1
THE VALUE OF THE OBJECTIVE FUNCTION IS:     62.000

THE TIME FOR BALAS WITH SENJU-TOYODA WAS   150 MILLISECONDS
```

2Ø.Ø	18.Ø	17.Ø	15.Ø	15.Ø	1Ø.Ø	5.Ø	3.Ø	1.Ø
3Ø.Ø	25.Ø	2Ø.Ø	18.Ø	17.Ø	11.Ø	5.Ø	2.Ø	1.Ø
6Ø.Ø								

THE RESULTS FROM SENJU-TOYODA ARE

```
X(  1)=   Ø
X(  2)=   Ø
X(  3)=   1
X(  4)=   Ø
X(  5)=   1
X(  6)=   1
X(  7)=   1
X(  8)=   1
X(  9)=   1
X( 1Ø)=   1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 52.ØØØ

THE TIME FOR SENJU-TOYODA WAS   Ø MILLISECONDS

↑/

THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE

```
X(  1)=   Ø
X(  2)=   Ø
X(  3)=   1
X(  4)=   1
X(  5)=   1
X(  6)=   Ø
X(  7)=   1
X(  8)=   Ø
X(  9)=   Ø
```

```
              X( 10)=    0
THE VALUE OF THE OBJECTIVE FUNCTION IS!      52.000

THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS  284 MILLISECONDS


↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/


THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE


              X(  1)=    0
              X(  2)=    0
              X(  3)=    1
              X(  4)=    0
              X(  5)=    1
              X(  6)=    1
              X(  7)=    1
              X(  8)=    1
              X(  9)=    1
              X( 10)=    1
THE VALUE OF THE OBJECTIVE FUNCTION IS!      52.000

THE TIME FOR BALAS WITH SENJU-TOYODA WAS   183 MILLISECONDS
```

```
    1.Ø      2.Ø      3.Ø      4.Ø

    4.Ø      3.Ø      2.Ø      1.Ø      4.Ø

    1.Ø      4.Ø      2.Ø      3.Ø      6.Ø


    THE RESULTS FROM SENJU-TOYODA ARE

                  X(   1)=    Ø
                  X(   2)=    Ø
                  X(   3)=    1
                  X(   4)=    1

THE VALUE OF THE OBJECTIVE FUNCTION IS:      7.ØØØ

THE TIME FOR SENJU-TOYODA WAS          17 MILLISECONDS

↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/

    THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE

                  X(   1)=    Ø
                  X(   2)=    Ø
                  X(   3)=    1
                  X(   4)=    1

THE VALUE OF THE OBJECTIVE FUNCTION IS:      7.ØØØ

THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS   83 MILLISECONDS
```

↑/

THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE

```
X(   1)=    0
X(   2)=    0
X(   3)=    1
X(   4)=    1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 7.000

THE TIME FOR BALAS WITH SENJU-TOYODA WAS   66 MILLISECONDS

2.0	3.0	4.0	6.0	
1.0	4.0	3.0	1.0	6.0
2.0	6.0	7.0	12.0	15.0
7.0	4.0	3.0	2.0	11.0

THE RESULTS FROM SENJU-TOYODA ARE

```
X(  1)=   1
X(  2)=   0
X(  3)=   0
X(  4)=   1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 8.000

THE TIME FOR SENJU-TOYODA WAS   .17 MILLISECONDS

↑/

THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE

```
X(  1)=   1
X(  2)=   0
X(  3)=   0
X(  4)=   1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 8.000

THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS 100 MILLISECONDS

↑/

THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE

```
X(   1)=    1
X(   2)=    0
X(   3)=    0
X(   4)=    1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 8.000

THE TIME FOR BALAS WITH SENJU-TOYODA WAS   67 MILLISECONDS

3.0	2.0	1.0	4.0	
1.0	2.0	4.0	1.0	5.0
2.0	5.0	1.0	1.0	6.0
6.0	1.0	2.0	2.0	10.0
1.0	1.0	1.0	1.0	3.0

THE RESULTS FROM SENJU-TOYODA ARE

```
X(   1)=    1
X(   2)=    0
X(   3)=    0
X(   4)=    1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 7.000

THE TIME FOR SENJU-TOYODA WAS 0 MILLISECONDS

↑/

THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE

```
X(   1)=    1
X(   2)=    0
X(   3)=    0
X(   4)=    1
```

```
THE VALUE OF THE OBJECTIVE FUNCTION IS:       7.000

THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS  116 MILLISECONDS

↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/

THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE

                  X(   1)=    1
                  X(   2)=    0
                  X(   3)=    0
                  X(   4)=    1

THE VALUE OF THE OBJECTIVE FUNCTION IS:       7.000

THE TIME FOR BALAS WITH SENJU-TOYODA WAS    83 MILLISECONDS
```

100.0	600.0	1200.0	2400.0	500.0	2000.0	
8.0	12.0	13.0	64.0	22.0	41.0	80.0
8.0	12.0	13.0	75.0	22.0	41.0	96.0
3.0	6.0	4.0	18.0	6.0	4.0	20.0
5.0	10.0	8.0	32.0	6.0	12.0	36.0
5.0	13.0	8.0	42.0	6.0	20.0	44.0
5.0	13.0	8.0	48.0	6.0	20.0	48.0
0.0	0.0	0.0	0.0	8.0	0.0	10.0
3.0	0.0	4.0	0.0	8.0	0.0	18.0
3.0	2.0	4.0	0.0	8.0	4.0	22.0
3.0	2.0	4.0	8.0	8.0	4.0	24.0

THE RESULTS FROM SENJU-TOYODA ARE

X(1)= 0

```
                    X(   2)=    1
                    X(   3)=    1
                    X(   4)=    0
                    X(   5)=    0
                    X(   6)=    1

THE VALUE OF THE OBJECTIVE FUNCTION IS:   3800.000

THE TIME FOR SENJU-TOYODA WAS          16 MILLISECONDS


↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/

THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE


                    X(   1)=    0
                    X(   2)=    1
                    X(   3)=    1
                    X(   4)=    0
                    X(   5)=    0
                    X(   6)=    1

THE VALUE OF THE OBJECTIVE FUNCTION IS:   3800.000

THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS  184 MILLISECONDS


↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/

THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE


                    X(   1)=    0
                    X(   2)=    1
                    X(   3)=    1
                    X(   4)=    0
                    X(   5)=    0
                    X(   6)=    1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 3800,000

THE TIME FOR BALAS WITH SENJU-TOYODA WAS 134 MILLISECONDS

```
 1.Ø    1.Ø    1.Ø    1.Ø    1.Ø    1.Ø    1.Ø    1.Ø    1.Ø

 1.Ø    2.Ø    3.Ø    4.Ø    1.Ø    1.Ø    1.Ø    2.Ø    2.Ø
 3.Ø

 1.Ø    1.Ø    1.Ø    2.Ø    1.Ø    5.Ø    5.Ø    6.Ø    7.Ø
 4.Ø

 3.Ø    4.Ø    3.Ø    4.Ø    4.Ø    2.Ø    2.Ø    3.Ø    3.Ø
1Ø.Ø

 6.Ø    8.Ø    8.Ø    9.Ø    6.Ø    5.Ø    5.Ø    5.Ø    5.Ø
2Ø.Ø

 3.Ø    4.Ø    4.Ø    4.Ø    3.Ø    7.Ø    9.Ø   1Ø.Ø   1Ø.Ø
1Ø.Ø


 THE RESULTS FROM SENJU-TOYODA ARE

           X(  1)=   1
           X(  2)=   Ø
           X(  3)=   Ø
           X(  4)=   Ø
           X(  5)=   1
           X(  6)=   Ø
           X(  7)=   Ø
           X(  8)=   Ø
           X(  9)=   Ø
           X( 1Ø)=   Ø
```

```
THE VALUE OF THE OBJECTIVE FUNCTION IS!      2.000

THE TIME FOR SENJU-TOYODA WAS          17 MILLISECONDS

↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/

THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE

                X(  1)=   0
                X(  2)=   1
                X(  3)=   0
                X(  4)=   0
                X(  5)=   1
                X(  6)=   0
                X(  7)=   0
                X(  8)=   0
                X(  9)=   0
                X( 10)=   0

THE VALUE OF THE OBJECTIVE FUNCTION IS!      2.000

THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS  333 MILLISECONDS

↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/

THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE

                X(  1)=   1
                X(  2)=   0
                X(  3)=   0
                X(  4)=   0
                X(  5)=   1
                X(  6)=   0
                X(  7)=   0
                X(  8)=   0
                X(  9)=   0
                X( 10)=   0
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 2.000

THE TIME FOR BALAS WITH SENJU-TOYODA WAS   100 MILLISECONDS

```
 1.0   1.0   1.0   1.0   1.0   1.0   1.0   1.0   1.0

 1.0   2.0   3.0   4.0   1.0   1.0   1.0   2.0   2.0
15.0

 1.0   1.0   1.0   2.0   1.0   5.0   5.0   6.0   7.0
25.0

 3.0   4.0   3.0   4.0   4.0   2.0   2.0   3.0   3.0
20.0

 6.0   8.0   8.0   9.0   6.0   5.0   5.0   5.0   5.0
35.0

 3.0   4.0   4.0   4.0   3.0   7.0   9.0  10.0  10.0
35.0
```

```
THE RESULTS FROM SENJU-TOYODA ARE

          X(  1)=   1
          X(  2)=   1
          X(  3)=   1
          X(  4)=   0
          X(  5)=   1
          X(  6)=   1
          X(  7)=   0
          X(  8)=   0
          X(  9)=   0
          X( 10)=   0
```

```
THE VALUE OF THE OBJECTIVE FUNCTION IS:      5.000

THE TIME FOR SENJU-TOYODA WAS           0 MILLISECONDS

↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/

THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE

                  X(   1)=    0
                  X(   2)=    1
                  X(   3)=    0
                  X(   4)=    1
                  X(   5)=    0
                  X(   6)=    1
                  X(   7)=    0
                  X(   8)=    1
                  X(   9)=    1
                  X(  10)=    0

THE VALUE OF THE OBJECTIVE FUNCTION IS:      5.000

THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS  3.5 SECONDS

↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/

THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE

                  X(   1)=    0
                  X(   2)=    1
                  X(   3)=    0
                  X(   4)=    1
                  X(   5)=    0
                  X(   6)=    1
                  X(   7)=    0
                  X(   8)=    1
                  X(   9)=    1
                  X(  10)=    0
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 5.000

THE TIME FOR BALAS WITH SENJU-TOYODA WAS   3.2 SECONDS

600.1	310.5	1800.0	3850.0	18.6	198.7	882.0	4200.0	402.5
20.0	5.0	100.0	200.0	2.0	4.0	60.0	150.0	80.0
450.0								
20.0	7.0	130.0	280.0	2.0	8.0	110.0	210.0	100.0
540.0								
60.0	3.0	50.0	100.0	4.0	2.0	20.0	40.0	6.0
200.0								
60.0	8.0	70.0	200.0	4.0	6.0	40.0	70.0	16.0
360.0								
60.0	13.0	70.0	250.0	4.0	10.0	60.0	90.0	20.0
440.0								
60.0	13.0	70.0	280.0	4.0	10.0	70.0	105.0	22.0
480.0								
5.0	2.0	20.0	100.0	2.0	5.0	10.0	60.0	0.0
200.0								
45.0	14.0	80.0	180.0	6.0	10.0	40.0	100.0	20.0
360.0								

```
  55.0    14.0    80.0   200.0     6.0    10.0    50.0   140.0    30.0
 440.0

  65.0    14.0    80.0   220.0     6.0    10.0    50.0   180.0    30.0
 480.0


   THE RESULTS FROM SENJU-TOYODA ARE

                X(   1)=    1
                X(   2)=    1
                X(   3)=    1
                X(   4)=    0
                X(   5)=    1
                X(   6)=    1
                X(   7)=    1
                X(   8)=    1
                X(   9)=    0
                X(  10)=    1

THE VALUE OF THE OBJECTIVE FUNCTION IS:   8336.900

THE TIME FOR SENJU-TOYODA WAS          17 MILLISECONDS


↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/


THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE

                X(   1)=    0
                X(   2)=    1
                X(   3)=    0
                X(   4)=    1
                X(   5)=    1
                X(   6)=    0
                X(   7)=    0
                X(   8)=    1
```

```
                X(  9)=    0
                X( 10)=    1

THE VALUE OF THE OBJECTIVE FUNCTION IS:   8706.100

THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS  366 MILLISECONDS


↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/


THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE

                X(  1)=    0
                X(  2)=    1
                X(  3)=    0
                X(  4)=    1
                X(  5)=    1
                X(  6)=    0
                X(  7)=    0
                X(  8)=    1
                X(  9)=    0
                X( 10)=    1

THE VALUE OF THE OBJECTIVE FUNCTION IS:   8706.100

THE TIME FOR BALAS WITH SENJU-TOYODA WAS   333 MILLISECONDS
```

100.0	220.0	90.0	400.0	300.0	400.0	205.0	120.0	160.0
400.0	140.0	100.0	1300.0	650.0				
8.0	4.0	13.0	80.0	70.0	80.0	45.0	15.0	28.0
130.0	32.0	20.0	120.0	40.0	550.0			
8.0	44.0	13.0	100.0	100.0	90.0	75.0	25.0	28.0
130.0	32.0	40.0	160.0	40.0	700.0			
3.0	6.0	4.0	20.0	20.0	30.0	8.0	3.0	12.0
40.0	6.0	3.0	20.0	5.0	130.0			
5.0	9.0	6.0	40.0	30.0	40.0	16.0	5.0	18.0
60.0	16.0	11.0	30.0	25.0	240.0			
5.0	11.0	7.0	50.0	40.0	40.0	19.0	7.0	18.0
70.0	21.0	17.0	30.0	25.0	280.0			
5.0	11.0	7.0	55.0	40.0	40.0	21.0	9.0	18.0
70.0	21.0	17.0	35.0	25.0	310.0			
0.0	0.0	1.0	10.0	4.0	10.0	0.0	6.0	0.0
32.0	3.0	0.0	70.0	10.0	110.0			
3.0	4.0	5.0	20.0	14.0	20.0	6.0	12.0	10.0
42.0	9.0	12.0	100.0	20.0	205.0			

3.0	6.0	9.0	30.0	29.0	20.0	12.0	12.0	10.0
42.0	18.0	18.0	110.0	20.0	260.0			

3.0	8.0	9.0	35.0	29.0	20.0	16.0	15.0	10.0
42.0	20.0	18.0	120.0	20.0	275.0			

THE RESULTS FROM SENJU-TOYODA ARE

```
X(  1)=  1
X(  2)=  1
X(  3)=  1
X(  4)=  0
X(  5)=  0
X(  6)=  1
X(  7)=  0
X(  8)=  1
X(  9)=  1
X( 10)=  1
X( 11)=  0
X( 12)=  1
X( 13)=  0
X( 14)=  1
X( 15)=  1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS!   3760.000

THE TIME FOR SENJU-TOYODA WAS   33 MILLISECONDS

↑/

THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE

```
X(  1)=   1
X(  2)=   1
X(  3)=   Ø
X(  4)=   1
X(  5)=   Ø
X(  6)=   1
X(  7)=   1
X(  8)=   Ø
X(  9)=   1
X( 1Ø)=   1
X( 11)=   Ø
X( 12)=   Ø
X( 13)=   Ø
X( 14)=   1
X( 15)=   1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 4Ø15.ØØØ

THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS 2.5 SECONDS

↑/

THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE

```
X(  1)=   1
X(  2)=   1
X(  3)=   Ø
X(  4)=   1
X(  5)=   Ø
X(  6)=   1
X(  7)=   1
X(  8)=   Ø
X(  9)=   1
X( 1Ø)=   1
X( 11)=   Ø
X( 12)=   Ø
X( 13)=   Ø
X( 14)=   1
X( 15)=   1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 4Ø15.ØØØ

THE TIME FOR BALAS WITH SENJU-TOYODA WAS   2.3 SECONDS

100.0	220.0	90.0	400.0	300.0	400.0	205.0	120.0	160.0
400.0	140.0	100.0	1300.0	650.0	320.0	480.0	80.0	60.0

8.0	4.0	13.0	80.0	70.0	80.0	45.0	15.0	28.0
130.0	32.0	20.0	120.0	40.0	30.0	20.0	6.0	3.0
550.0								

8.0	44.0	13.0	100.0	100.0	90.0	75.0	25.0	28.0
130.0	32.0	40.0	160.0	40.0	60.0	55.0	10.0	6.0
700.0								

3.0	6.0	4.0	20.0	20.0	30.0	8.0	3.0	12.0
40.0	6.0	3.0	20.0	5.0	0.0	5.0	3.0	0.0
130.0								

5.0	9.0	6.0	40.0	30.0	40.0	16.0	5.0	18.0
60.0	16.0	11.0	30.0	25.0	10.0	13.0	5.0	1.0
240.0								

5.0	11.0	7.0	50.0	40.0	40.0	19.0	7.0	18.0
70.0	21.0	17.0	30.0	25.0	15.0	25.0	5.0	1.0
280.0								

5.0	11.0	7.0	55.0	40.0	40.0	21.0	9.0	18.0
70.0	21.0	17.0	35.0	25.0	20.0	25.0	5.0	2.0
310.0								

Ø.Ø	Ø.Ø	1.Ø	1Ø.Ø	4.Ø	1Ø.Ø	Ø.Ø	6.Ø	Ø.Ø
32.Ø	3.Ø	Ø.Ø	7Ø.Ø	1Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
11Ø.Ø								
3.Ø	4.Ø	5.Ø	2Ø.Ø	14.Ø	2Ø.Ø	6.Ø	12.Ø	1Ø.Ø
42.Ø	9.Ø	12.Ø	1ØØ.Ø	2Ø.Ø	5.Ø	6.Ø	4.Ø	1.Ø
2Ø5.Ø								
3.Ø	6.Ø	9.Ø	3Ø.Ø	29.Ø	2Ø.Ø	12.Ø	12.Ø	1Ø.Ø
42.Ø	18.Ø	18.Ø	11Ø.Ø	2Ø.Ø	15.Ø	18.Ø	7.Ø	2.Ø
26Ø.Ø								
3.Ø	8.Ø	9.Ø	35.Ø	29.Ø	2Ø.Ø	16.Ø	15.Ø	1Ø.Ø
42.Ø	2Ø.Ø	18.Ø	12Ø.Ø	2Ø.Ø	2Ø.Ø	22.Ø	7.Ø	3.Ø
275.Ø								

THE RESULTS FROM SENJU-TOYODA ARE

```
X(  1)=   1
X(  2)=   1
X(  3)=   1
X(  4)=   Ø
X(  5)=   Ø
X(  6)=   Ø
X(  7)=   Ø
X(  8)=   Ø
X(  9)=   1
X( 1Ø)=   Ø
X( 11)=   Ø
X( 12)=   Ø
X( 13)=   Ø
```

```
X( 14)=   1
X( 15)=   1
X( 16)=   1
X( 17)=   1
X( 18)=   1
X( 19)=   1
X( 20)=   1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 6010.000

THE TIME FOR SENJU-TOYODA WAS   .33 MILLISECONDS

↑/

THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE

```
X(  1)=   1
X(  2)=   0
X(  3)=   0
X(  4)=   0
X(  5)=   0
X(  6)=   0
X(  7)=   0
X(  8)=   0
X(  9)=   0
X( 10)=   1
X( 11)=   0
X( 12)=   0
X( 13)=   0
X( 14)=   1
X( 15)=   1
X( 16)=   1
X( 17)=   1
X( 18)=   1
X( 19)=   1
X( 20)=   1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 6120.000

THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS 34.8 SECONDS

↑/

THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE

```
X(  1)=   1
X(  2)=   0
X(  3)=   0
X(  4)=   0
X(  5)=   0
X(  6)=   0
X(  7)=   0
X(  8)=   0
X(  9)=   0
X( 10)=   1
X( 11)=   0
X( 12)=   0
X( 13)=   0
X( 14)=   1
X( 15)=   1
X( 16)=   1
X( 17)=   1
X( 18)=   1
X( 19)=   1
X( 20)=   1
```

THE VALUE OF THE OBJECTIVE FUNCTION IS: 6120.000

THE TIME FOR BALAS WITH SENJU-TOYODA WAS 32.1 SECONDS

1.Ø	1.Ø	1.Ø	1.Ø	1.Ø	1.Ø	1.Ø	1.Ø	1.Ø
1.Ø	1.Ø	1.Ø	1.Ø	1.Ø	1.Ø	1.Ø	1.Ø	1.Ø
1.Ø	1.Ø	1.Ø	1.Ø	1.Ø				
Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	1.Ø	1.Ø	1.Ø	1.Ø	1.Ø
1.Ø	1.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1Ø.Ø			
Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	1.Ø	1.Ø	1.Ø	1.Ø	1.Ø
1.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	9.Ø			
Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	1.Ø	1.Ø	1.Ø	1.Ø	1.Ø
1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	9.Ø			
1.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	1.Ø	Ø.Ø
Ø.Ø	1.Ø	1.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	8.Ø			
Ø.Ø	1.Ø	Ø.Ø	1.Ø	Ø.Ø	1.Ø	1.Ø	Ø.Ø	1.Ø
1.Ø	Ø.Ø	1.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
1.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	9.Ø			

Ø.Ø	1.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	1.Ø	1.Ø	1.Ø
1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø
Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	8.Ø			
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	1.Ø	Ø.Ø
Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	1.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø
1.Ø	1.Ø	1.Ø	Ø.Ø	Ø.Ø	8.Ø			
Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	1.Ø	Ø.Ø
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø
1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	7.Ø			
Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	1.Ø	Ø.Ø	1.Ø
1.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	7.Ø			
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	1.Ø	Ø.Ø
Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	6.Ø			
1.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	1.Ø	Ø.Ø
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	6.Ø			
1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø
Ø.Ø	1.Ø	1.Ø	Ø.Ø	Ø.Ø	1.Ø	1.Ø	Ø.Ø	Ø.Ø
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	6.Ø			
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø
Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	4.Ø			
1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1.Ø	1.Ø	Ø.Ø	Ø.Ø
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	4.Ø			
1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø
Ø.Ø	1.Ø	1.Ø	Ø.Ø	1.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	4.Ø			
Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1.Ø	Ø.Ø
Ø.Ø	1.Ø	Ø.Ø	Ø.Ø	Ø.Ø	1.Ø	1.Ø	Ø.Ø	Ø.Ø

0.0	0.0	0.0	0.0	0.0	4.0			
0.0	1.0	0.0	1.0	0.0	0.0	0.0	0.0	1.0
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	4.0			
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	1.0	1.0	0.0	0.0	0.0
0.0	0.0	1.0	0.0	0.0	4.0			
0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
0.0	0.0	1.0	0.0	1.0	1.0	0.0	0.0	0.0
0.0	1.0	0.0	0.0	0.0	4.0			
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	1.0	0.0	1.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	3.0			
1.0	0.0	1.0	0.0	0.0	1.0	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	3.0			
0.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
1.0	0.0	0.0	0.0	0.0	3.0			
1.0	0.0	1.0	0.0	0.0	1.0	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	3.0			
0.0	0.0	0.0	0.0	0.0	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	0.0	0.0	2.0			
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	1.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	2.0			
0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	2.0			
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	1.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	2.0			
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	1.0	1.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	2.0			
0.0	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	1.0	0.0	0.0	0.0
1.0	0.0	0.0	0.0	0.0	2.0			
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
1.0	1.0	0.0	0.0	0.0	2.0			
0.0	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	1.0			
0.0	0.0	0.0	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	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	1.0	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      1.0      1.0      0.0      0.0      0.0      0.0      0.0      0.0
     0.0      0.0      0.0      0.0      0.0      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      1.0      0.0      1.0      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
     1.0      0.0      1.0      0.0      0.0      1.0
     THE RESULTS FROM SENJU-TOYODA ARE
                    X(  1)=   1
                    X(  2)=   1
                    X(  3)=   1
                    X(  4)=   1
                    X(  5)=   1
                    X(  6)=   0
                    X(  7)=   1
                    X(  8)=   1
                    X(  9)=   1
                    X( 10)=   1
                    X( 11)=   1
                    X( 12)=   0
                    X( 13)=   1
                    X( 14)=   1
                    X( 15)=   0
                    X( 16)=   0
                    X( 17)=   1
                    X( 18)=   1
                    X( 19)=   1
                    X( 20)=   0
                    X( 21)=   0
                    X( 22)=   1
                    X( 23)=   1
                    X( 24)=   1
                    X( 25)=   1
THE VALUE OF THE OBJECTIVE FUNCTION IS!      19.000
THE TIME FOR SENJU-TOYODA WAS              150 MILLISECONDS
↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/
THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE
                    X(  1)=   1
                    X(  2)=   1
                    X(  3)=   1
                    X(  4)=   0
                    X(  5)=   1
                    X(  6)=   0
                    X(  7)=   1
                    X(  8)=   1
                    X(  9)=   1
                    X( 10)=   1
                    X( 11)=   1
```

```
                 X( 12)=    Ø
                 X( 13)=    1
                 X( 14)=    1
                 X( 15)=    Ø
                 X( 16)=    1
                 X( 17)=    Ø
                 X( 18)=    1
                 X( 19)=    1
                 X( 2Ø)=    1
                 X( 21)=    Ø
                 X( 22)=    1
                 X( 23)=    1
                 X( 24)=    1
                 X( 25)=    1
THE VALUE OF THE OBJECTIVE FUNCTION IS:     19,ØØØ
THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS 32.8 SECONDS
↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/
THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE
                 X(  1)=    1
                 X(  2)=    1
                 X(  3)=    1
                 X(  4)=    Ø
                 X(  5)=    1
                 X(  6)=    Ø
                 X(  7)=    1
                 X(  8)=    1
                 X(  9)=    1
                 X( 1Ø)=    1
                 X( 11)=    1
                 X( 12)=    Ø
                 X( 13)=    1
                 X( 14)=    1
                 X( 15)=    Ø
                 X( 16)=    1
                 X( 17)=    Ø
                 X( 18)=    1
                 X( 19)=    1
                 X( 2Ø)=    1
                 X( 21)=    Ø
                 X( 22)=    1
                 X( 23)=    1
                 X( 24)=    1
                 X( 25)=    1
THE VALUE OF THE OBJECTIVE FUNCTION IS:     19,ØØØ
THE TIME FOR BALAS WITH SENJU-TOYODA WAS  31.4 SECONDS
```

100.0	220.0	90.0	400.0	300.0	400.0	205.0	120.0	160.0
400.0	140.0	100.0	1300.0	650.0	320.0	480.0	80.0	60.0
3100.0	1100.0	950.0	450.0	300.0	220.0	200.0	520.0	
8.0	4.0	13.0	80.0	70.0	80.0	45.0	15.0	28.0
130.0	32.0	20.0	120.0	40.0	30.0	20.0	6.0	3.0
220.0	50.0	30.0	50.0	12.0	5.0	8.0	18.0	930.0
8.0	44.0	13.0	100.0	100.0	90.0	75.0	25.0	28.0
130.0	32.0	40.0	160.0	40.0	60.0	55.0	10.0	6.0
290.0	80.0	90.0	70.0	27.0	17.0	8.0	28.0	1210.0
3.0	6.0	4.0	20.0	20.0	30.0	8.0	3.0	12.0
40.0	6.0	3.0	20.0	5.0	0.0	5.0	3.0	0.0
30.0	40.0	10.0	0.0	5.0	0.0	0.0	10.0	272.0
5.0	9.0	6.0	40.0	30.0	40.0	16.0	5.0	18.0
60.0	16.0	11.0	30.0	25.0	10.0	13.0	5.0	1.0
60.0	50.0	20.0	30.0	10.0	5.0	3.0	20.0	426.0
5.0	11.0	7.0	50.0	40.0	40.0	19.0	7.0	18.0
70.0	21.0	17.0	30.0	25.0	15.0	25.0	5.0	1.0
70.0	55.0	20.0	50.0	15.0	15.0	6.0	20.0	532.0
5.0	11.0	7.0	55.0	40.0	40.0	21.0	9.0	18.0
70.0	21.0	17.0	35.0	25.0	20.0	25.0	5.0	2.0
70.0	55.0	20.0	50.0	20.0	15.0	6.0	20.0	572.0
0.0	0.0	1.0	10.0	4.0	10.0	0.0	6.0	0.0
32.0	3.0	0.0	70.0	10.0	0.0	0.0	0.0	0.0
30.0	10.0	0.0	10.0	10.0	5.0	0.0	10.0	240.0
3.0	4.0	5.0	20.0	14.0	20.0	6.0	12.0	10.0
42.0	9.0	12.0	100.0	20.0	5.0	6.0	4.0	1.0
50.0	30.0	5.0	20.0	20.0	10.0	10.0	20.0	400.0
3.0	6.0	9.0	30.0	29.0	20.0	12.0	12.0	10.0
42.0	18.0	18.0	110.0	20.0	15.0	18.0	7.0	2.0
60.0	50.0	25.0	25.0	25.0	15.0	10.0	28.0	470.0
3.0	8.0	9.0	35.0	29.0	20.0	16.0	15.0	10.0
42.0	20.0	18.0	120.0	20.0	20.0	22.0	7.0	3.0
60.0	55.0	25.0	30.0	25.0	15.0	10.0	28.0	490.0

```
THE RESULTS FROM SENJU-TOYODA ARE
            X(  1)=   1
            X(  2)=   1
            X(  3)=   1
            X(  4)=   0
            X(  5)=   0
            X(  6)=   0
            X(  7)=   0
            X(  8)=   0
            X(  9)=   1
            X( 10)=   0
            X( 11)=   0
            X( 12)=   0
            X( 13)=   0
            X( 14)=   1
            X( 15)=   1
            X( 16)=   1
            X( 17)=   1
```

```
                    X( 18)=    1
                    X( 19)=    1
                    X( 2Ø)=    1
                    X( 21)=    1
                    X( 22)=    1
                    X( 23)=    1
                    X( 24)=    Ø
                    X( 25)=    1
                    X( 26)=    1
                    X( 27)=    1
                    X( 28)=    1
THE VALUE OF THE OBJECTIVE FUNCTION IS:  124ØØ.ØØØ
THE TIME FOR SENJU-TOYODA WAS          47 MILLISECONDS
↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/
THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE
                    X(  1)=    1
                    X(  2)=    1
                    X(  3)=    1
                    X(  4)=    Ø
                    X(  5)=    Ø
                    X(  6)=    Ø
                    X(  7)=    Ø
                    X(  8)=    Ø
                    X(  9)=    1
                    X( 1Ø)=    Ø
                    X( 11)=    Ø
                    X( 12)=    Ø
                    X( 13)=    Ø
                    X( 14)=    1
                    X( 15)=    1
                    X( 16)=    1
                    X( 17)=    1
                    X( 18)=    1
                    X( 19)=    1
                    X( 2Ø)=    1
                    X( 21)=    1
                    X( 22)=    1
                    X( 23)=    1
                    X( 24)=    Ø
                    X( 25)=    1
                    X( 26)=    1
                    X( 27)=    1
                    X( 28)=    1
THE VALUE OF THE OBJECTIVE FUNCTION IS:  124ØØ.ØØØ
THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS 457.1 SECONDS
↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/
THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE
                    X(  1)=    1
                    X(  2)=    1
                    X(  3)=    1
                    X(  4)=    Ø
```

```
                X(   5)=    0
                X(   6)=    0
                X(   7)=    0
                X(   8)=    0
                X(   9)=    1
                X(  10)=    0
                X(  11)=    0
                X(  12)=    0
                X(  13)=    0
                X(  14)=    1
                X(  15)=    1
                X(  16)=    1
                X(  17)=    1
                X(  18)=    1
                X(  19)=    1
                X(  20)=    1
                X(  21)=    1
                X(  22)=    1
                X(  23)=    1
                X(  24)=    0
                X(  25)=    1
                X(  26)=    1
                X(  27)=    1
                X(  28)=    1
THE VALUE OF THE OBJECTIVE FUNCTION IS:  12400.000
THE TIME FOR BALAS WITH SENJU-TOYODA WAS 428.2 SECONDS
```

56Ø.Ø	1125.Ø	3ØØ.Ø	62Ø.Ø	21ØØ.Ø	431.Ø	68.Ø	328.Ø	47.Ø
322.Ø	196.Ø	41.Ø	25.Ø	425.Ø	426Ø.Ø	416.Ø	115.Ø	82.Ø
631.Ø	132.Ø	42Ø.Ø	86.Ø	42.Ø	1Ø3.Ø	215.Ø	81.Ø	91.Ø
49.Ø	42Ø.Ø	316.Ø	72.Ø	71.Ø	49.Ø	1Ø8.Ø	116.Ø	9Ø.Ø
4Ø.Ø	91.Ø	1Ø.Ø	3Ø.Ø	16Ø.Ø	2Ø.Ø	3.Ø	12.Ø	3.Ø
9.Ø	25.Ø	1.Ø	1.Ø	1Ø.Ø	28Ø.Ø	1Ø.Ø	8.Ø	1.Ø
49.Ø	8.Ø	21.Ø	6.Ø	1.Ø	5.Ø	1Ø.Ø	8.Ø	2.Ø
Ø.Ø	1Ø.Ø	42.Ø	6.Ø	4.Ø	8.Ø	Ø.Ø	1Ø.Ø	1.Ø
16.Ø	92.Ø	41.Ø	16.Ø	15Ø.Ø	23.Ø	4.Ø	18.Ø	6.Ø
12.Ø	8.Ø	2.Ø	1.Ø	Ø.Ø	2ØØ.Ø	2Ø.Ø	6.Ø	2.Ø
7Ø.Ø	9.Ø	22.Ø	4.Ø	1.Ø	5.Ø	1Ø.Ø	6.Ø	4.Ø
4.Ø	12.Ø	8.Ø	4.Ø	3.Ø	Ø.Ø	1Ø.Ø	Ø.Ø	6.Ø
3.Ø	Ø.Ø	39.Ø	32.Ø	71.Ø	8Ø.Ø	26.Ø	5.Ø	4Ø.Ø
12.Ø	3Ø.Ø	15.Ø	Ø.Ø	1.Ø	23.Ø	1ØØ.Ø	Ø.Ø	2Ø.Ø
4Ø.Ø	6.Ø	8.Ø	Ø.Ø	6.Ø	4.Ø	22.Ø	4.Ø	6.Ø
5.Ø	14.Ø	8.Ø	2.Ø	8.Ø	Ø.Ø	2Ø.Ø	Ø.Ø	Ø.Ø
8.Ø	71.Ø	3Ø.Ø	6Ø.Ø	2ØØ.Ø	18.Ø	6.Ø	3Ø.Ø	4.Ø
31.Ø	6.Ø	3.Ø	Ø.Ø	18.Ø	6Ø.Ø	21.Ø	4.Ø	Ø.Ø
32.Ø	15.Ø	31.Ø	2.Ø	2.Ø	7.Ø	8.Ø	2.Ø	8.Ø
2.Ø	8.Ø	6.Ø	7.Ø	1.Ø	Ø.Ø	Ø.Ø	2Ø.Ø	8.Ø
3Ø.Ø	52.Ø	3Ø.Ø	42.Ø	17Ø.Ø	9.Ø	7.Ø	2Ø.Ø	Ø.Ø
21.Ø	4.Ø	1.Ø	2.Ø	14.Ø	31Ø.Ø	8.Ø	4.Ø	6.Ø
18.Ø	15.Ø	38.Ø	1Ø.Ø	4.Ø	8.Ø	6.Ø	Ø.Ø	Ø.Ø
Ø.Ø	1Ø.Ø	6.Ø	1.Ø	3.Ø	Ø.Ø	3.Ø	5.Ø	4.Ø

```
 THE RESULTS FROM SENJU-TOYODA ARE
            X(  1)=   1
            X(  2)=   Ø
            X(  3)=   Ø
            X(  4)=   1
            X(  5)=   Ø
            X(  6)=   1
            X(  7)=   Ø
            X(  8)=   1
            X(  9)=   Ø
            X( 1Ø)=   1
            X( 11)=   1
            X( 12)=   1
            X( 13)=   Ø
            X( 14)=   1
            X( 15)=   1
            X( 16)=   1
            X( 17)=   1
            X( 18)=   1
            X( 19)=   1
            X( 2Ø)=   1
            X( 21)=   Ø
            X( 22)=   Ø
            X( 23)=   1
            X( 24)=   1
            X( 25)=   1
            X( 26)=   1
```

```
                    X( 27)=    1
                    X( 28)=    1
                    X( 29)=    1
                    X( 30)=    1
                    X( 31)=    1
                    X( 32)=    1
                    X( 33)=    1
                    X( 34)=    1
                    X( 35)=    1
                    X( 36)=    1
                    X( 37)=    1
                    X( 38)=    1
                    X( 39)=    1
THE VALUE OF THE OBJECTIVE FUNCTION IS:  10279.000
THE TIME FOR SENJU-TOYODA WAS          63 MILLISECONDS
↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/
THE RESULTS FROM IMPLICIT ENUMERATION W/OUT SENJU-TOYODA ARE
                    X(  1)=    1
                    X(  2)=    1
                    X(  3)=    0
                    X(  4)=    1
                    X(  5)=    0
                    X(  6)=    1
                    X(  7)=    0
                    X(  8)=    1
                    X(  9)=    1
                    X( 10)=    0
                    X( 11)=    1
                    X( 12)=    0
                    X( 13)=    1
                    X( 14)=    0
                    X( 15)=    1
                    X( 16)=    1
                    X( 17)=    1
                    X( 18)=    1
                    X( 19)=    1
                    X( 20)=    1
                    X( 21)=    0
                    X( 22)=    0
                    X( 23)=    1
                    X( 24)=    0
                    X( 25)=    1
                    X( 26)=    0
                    X( 27)=    1
                    X( 28)=    1
                    X( 29)=    1
                    X( 30)=    0
                    X( 31)=    1
                    X( 32)=    1
                    X( 33)=    0
                    X( 34)=    1
```

```
                 X( 35)=   1
                 X( 36)=   1
                 X( 37)=   1
                 X( 38)=   1
                 X( 39)=   1
THE VALUE OF THE OBJECTIVE FUNCTION IS:  10618.000
THE TIME FOR BALAS W/OUT SENJU-TOYODA WAS 2934 SECONDS
↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/↑/
THE RESULTS FROM IMPLICIT ENUMERATION WITH SENJU-TOYODA ARE
                 X(  1)=   1
                 X(  2)=   1
                 X(  3)=   0
                 X(  4)=   1
                 X(  5)=   0
                 X(  6)=   1
                 X(  7)=   0
                 X(  8)=   1
                 X(  9)=   1
                 X( 10)=   0
                 X( 11)=   1
                 X( 12)=   0
                 X( 13)=   1
                 X( 14)=   0
                 X( 15)=   1
                 X( 16)=   1
                 X( 17)=   1
                 X( 18)=   1
                 X( 19)=   1
                 X( 20)=   1
                 X( 21)=   0
                 X( 22)=   0
                 X( 23)=   1
                 X( 24)=   0
                 X( 25)=   1
                 X( 26)=   0
                 X( 27)=   1
                 X( 28)=   1
                 X( 29)=   1
                 X( 30)=   0
                 X( 31)=   1
                 X( 32)=   1
                 X( 33)=   0
                 X( 34)=   1
                 X( 35)=   1
                 X( 36)=   1
                 X( 37)=   1
                 X( 38)=   1
                 X( 39)=   1
THE VALUE OF THE OBJECTIVE FUNCTION IS:  10618.000
THE TIME FOR BALAS WITH SENJU-TOYODA WAS 2720 SECONDS
```