T-1470

ON NOT SEARCHING FOR THE MULTIPLIER IN KNAPSACK PROBLEMS

BY

David A. Gulley

ProQuest Number: 10781786

All rights reserved

INFORMATION TO ALL USERS
The quality of this reproduction is dependent upon the quality of the copy submitted.

In the unlikely event that the author did not send a complete manuscript
and there are missing pages, these will be noted. Also, if material had to be removed,
a note will indicate the deletion.

ProQuest 10781786

Published by ProQuest LLC (2018). Copyright of the Dissertation is held by the Author.

All rights reserved.
This work is protected against unauthorized copying under Title 17, United States Code
Microform Edition © ProQuest LLC.

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 – 1346

T-1470

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 (Mathematics).

Signed: David H. Fulley
Student

Golden, Colorado
Date: April 18, 1972

ARTHUR LAKES LIBRARY
COLORADO SCHOOL OF MINES
GOLDEN, COLORADO

Approved: Robert H. Dorroh
Thesis Advisor

Joseph R. Lee
Head of Department

Golden, Colorado
Date: April 19, 1972

ABSTRACT

The Slippery Algorithm is a simple-to-use and very fast technique for the solution of knapsack problems. While the method does not guarantee an optimum allocation, preliminary computational results indicate that very few problems are solved to less than 98 per cent of optimality. The key feature of the Slippery Algorithm is its use of the ratios of objective-function coefficient to constraint-function coefficient; when using the Slippery Algorithm, the higher a particular ratio is relative to other ratios, the stronger the likelihood that the variable will be upper-bounded in the final allocation. The use of these ratios is not only intuitively appealing, but is shown to be equivalent to the use of multipliers in Everett's Generalized LaGrange Multiplier solution technique. Of the 100 knapsack problems solved, enumerative methods were not frequently needed to determine the optimum allocation. For the problems solved, the optimum integer-variable solution is quite close to the optimum continuous-variable solution.

CONTENTS

	Page
Abstract	iii
Introduction	1
Solving Knapsack Problems via "Critical Values" of the Multiplier	4
A GLM Approach to the Knapsack Problem	4
The Slippery Algorithm	8
Computational Results	11
Conclusions	19
Appendix	21
Bibliography	22

ILLUSTRATIONS AND TABLES

	Page
Display 1	9
Table 1	13
Figure 1	16
Figure 2	17

88 Ishiro

INTRODUCTION

The knapsack problem, a hypothetical situation with many real-world overtones, can be posed as follows: suppose an ardent backpacker is contemplating an extended trip into the back country. While his knapsack is large enough to accomodate any number of items, he feels that he should limit the total weight of the pack's contents. There are many items available for inclusion in the pack, each with a certain weight and a certain level of utility or usefulness. Given a list of such objects, and the maximum allowable total weight, he wishes to select those items for inclusion which maximize total utility without violating the weight limitation. Mathematically, this can be stated as:

$$\text{Maximize} \quad H(x)=\sum_{i=1}^{n} H_i x_i$$

$$(1) \qquad \text{Subject to} \quad C(x)=\sum_{i=1}^{n} C_i x_i \leq c$$

$$x_i=0 \text{ or } 1,\ H_i>0, \text{ and } C_i>0 \text{ for } i=1,\ldots,n$$

where H_i is the utility of the ith item, C_i is the weight of the ith item, c is the limitation on total weight, and $x_i=0$ if the ith item is rejected and $x_i=1$ if the ith item is included in the pack.

While the problem as stated is imaginary, analogous real-world problems are legion. Examples include capital budgeting, personnel assignment, and many other problems involving yes-or-no decisions based on the limits of a single resource.

The knapsack problem was first presented in the literature by one of the pioneers of mathematical programming, George B. Dantzig (1957). Dantzig noted that an optimum solution to the knapsack problem could be arrived at via linear programming (LP) if the variables were continuous between 0 and 1, but that one could not guarantee optimality for binary

variable problems when using LP. In his paper, Dantzig stated that the earliest method of guaranteeing an optimum solution was dynamic programming, but he also described a heuristic solution approach, which follows.

Graph the coefficients of the variables on the Cartesian coordinates as the points (C_i, H_i). Starting in the vertical position, rotate a chord clockwise toward the positive x-axis. As the chord passes the points (C_i, H_i), accumulate the C_i's. Stop when the addition of the next C_i would cause the sum to exceed the right-hand-side, c. Set the x_i's above the chord (the points passed by the chord) equal to 1; set the rest of the variables equal to 0. Note that this method sets the variables to their upper bound in order of decreasing value of the ratio H_i/C_i. Many knapsack algorithms which have since been developed have also in some way used these ratios (which shall herein be termed the "efficiency" of a variable). Other methods employing these ratios include those of Gilmore and Gomory (1963), Senju and Toyoda (1968), and Greenberg and Hegerich (1970).

Dantzig showed that the solution to the related linear programming problem (i.e., variables continuous on [0,1]) is directly dependent on the efficiencies of the variables; if the variables are subscripted from the most efficient (i=1) to the least efficient (i=n), then the LP solution is:

$$
\begin{aligned}
&x_i=1 \text{ if } i<r \\
(2) \qquad &x_i=0 \text{ if } i>r \\
&x_r=(c-\sum_{i=1}^{r-1} C_i)/C_r
\end{aligned}
$$

where r is the least integer $(1 \leq r \leq n)$ for which $\sum_{i=1}^{r} C_i \geq c$. If $x_r=0$, the optimal solution to the continuous-variable problem is the optimal solution to the discrete-variable problem.

Because of the close relationship between variable efficiencies and the optimum LP allocation, solution techniques to the knapsack problem are most often compared to linear programming. However, it will be shown in this work that at least one such method, the Slippery Algorithm, is directly analogous to the Generalized LaGrange Multiplier (GLM) technique

of Everett (1963), and therefore the Slippery Algorithm has the same theoretical justification as GLM.

As the knapsack problem reflects many highly restrictive conditions compared to a general allocation problem (linearity of objective and constraining functions, independent binary variables, and a single constraint), it might seem reasonable to suppose that many excellent techniques must exist for determining an optimal solution. However, both the problem's ubiquity and the large size of some samples suggest that greater computational efficiency be sought. Furthermore, knapsack problems occur in environments devoid of digital computers, and thus a method simple enough for hand solution is desirable. The Slippery Algorithm offers the desired speed and simplicity of solution, although sometimes in lieu of an optimal answer.

It is the purpose of this study to show the relationship between GLM and the Slippery Algorithm, to determine the computational speed of the Slippery Algorithm, and to compare the goodness of the algorithm's answer with the optimum allocation.

As is surely the case with most such endeavors, the author and his work have both benefited from the exchange of ideas among many people, certainly from all members of his thesis committee. The author wishes to particularly acknowledge the aid of Mr. C. O. Baer, instructor of mathematics, in the development and running of the computer programs necessary for this investigation, and of Dr. H. S. Swanson, assistant professor of mathematics and originator of the Slippery Algorithm, whose contributions to the project most definitely went beyond any commitment a reasonable man could have expected.

SOLVING KNAPSACK PROBLEMS VIA "CRITICAL VALUES" OF THE MULTIPLIER

A GLM Approach to the Knapsack Problem

As shown in the introduction, the linear programming solution to (1) is closely related to the ratios of objective coefficient to constraint coefficient, and thus solution techniques for the binary-variable case which rely on efficiencies are reminiscent of linear programming. It will now be shown that such procedures for (1) are also related to the Generalized LaGrange Multiplier technique, and that in fact the variable efficiencies correspond to special values of the GLM multiplier which are in more general cases determined by trial-and-error methods.

The main theorem of Everett (1963) states that for a given value of the multiplier $\lambda \geq 0$, the solution x^* which maximizes the Lagrangian $H(x) - \lambda C(x)$ is also the solution which maximizes $H(x)$ for that constrained problem where $c = C(x^*)$ (and where $C(x^*)$ may or may not be the particular right-hand-side of interest). Thus an optimum solution is obtained for a problem which duplicates the original problem except possibly for the amount of resource available. This quantity of resource, $C(x^*)$, is instead determined by the solution itself.

The solution vector x^* is the n-tuple of optimum values of $x_1, \ldots, x_n$. While there is no one method which will find that n-tuple which maximizes the Lagrangian resulting from generalized kinds of problems, in the case of the knapsack problem the maximization of the Lagrangian merely involves the maximization of n subproblems, namely $(H_i - \lambda C_i)x_i$, for each i. As the x_i's are binary variables, these subproblems are maximized by setting $x_i=0$ if the coefficient $(H - \lambda C_i)$ is negative, and setting $x_i=1$ if the coefficient is positive. If the coefficient is zero, it

indicates a special case to be discussed in more detail later.

Although GLM offers great promise for certain types of problems, selection of the appropriate multiplier values can, in general, be troublesome. Fortunately, for the knapsack problem, the values of λ which result in allocation changes (changes in $C(x^*)$) are the values of the variable efficiencies, H_i/C_i Such a value of the multiplier causes at least one coefficient, say $(H_i - \lambda C_i)$ to be equal to zero:

$$(H_i - \lambda C_i) = H_i - (H_i/C_i)C_i = 0.$$

These values of the multiplier are "critical values" in the sense that these values, and only these values, produce a change in the optimal solution to the Lagrangian for the knapsack problem. To either side of a critical value, at least one coefficient $(H_i - \lambda C_i)$ changes in sign. Thus the value of x_i, and therefore x^*, changes so that $H(x^*)$ and $C(x^*)$ change.

The relationship between an optimum solution to (1) and critical values of the multiplier shall presently be shown more formally, but first a special case of the main theorem of Everett is restated, which will indicate the role of the Lagrangian in determining an optimum allocation x^* to the knapsack problem (1). For the remainder of this thesis it will be assumed that the variables have been reordered so that the efficiencies decrease as the variable subscript increases. For ease of presentation of Theorem 1, a "strategy set" S is defined which consists of all possible strategies for (1).

<u>Theorem 1</u>: Let λ be a non-negative real number; if $x^* \varepsilon S$ maximizes the function $H(x) - \lambda C(x)$ over all $x \varepsilon S$, then x^* maximizes $H(x)$ over all those $x \varepsilon S$ such that $C(x) \leq C(x^*)$.

<u>Proof</u>: By assumption, x^* is a strategy (not necessarily unique) which maximizes $H(x)-\lambda C(x)$ over all $x \varepsilon S$. Therefore, for all $x \varepsilon S$, $H(x^*)-\lambda C(x^*) \geq H(x)-\lambda C(x)$, and thus $H(x^*) \geq H(x) + \lambda[C(x^*)-C(x)]$ for all $x \varepsilon S$. As this latter inequality is true for all $x \varepsilon S$, it holds for any subset of S, and in particular, it holds for that subset S' of S containing those strategies x' such that $C(x') \leq C(x^*)$. Thus for these strategies, $C(x^*)-C(x') \geq 0$, and as λ is non-negative by definition, the term

$\lambda[C(x^*)-C(x')]$ is non-negative, and our inequality reduces to $H(x^*)\geq H(x')$ for all $x\varepsilon S$, and the theorem is proved.

Theorem 2: Given the knapsack problem (1) and x_i, $i=1,\ldots,n$ such that $H_i/C_i\geq H_j/C_j$ whenever $i<j$; then if $\lambda\geq 0$ is fixed at a non-critical value, there exists a positive integer $k\leq n$ such that the Lagrangian resulting from (1) is maximized by setting the first k variables $(x_1,\ldots,x_k)$ equal to 1 and the remaining variables $(x_{k+1},\ldots,x_n)$ equal to 0.

Proof: Suppose the conclusion is false. Then there exists a solution $\hat{x}$ maximizing the Lagrangian resulting from (1) which has the property that $\hat{x}$ includes components x_i and x_j such that $i<j$, $x_j=1$, and $x_i=0$. Thus $(H_i-\lambda C_i)<0$ and $(H_j-C_j)>0$. This implies that $H_i/C_i<\lambda<H_j/C_j$. This is a contradiction, since $i<j$ implies that $H_i/C_i>H_j/C_j$. Thus the theorem is proved.[1/]

This theorem states that for (1), when the variables are ordered by decreasing efficiency, there is a series of optimal allocations, $[x^*]_{k=1}^{n}$ each of which is $x_1=\ldots=x_k=1$, $x_{k+1}=\ldots=x_n=0$, which corresponds to optimal solutions for a series of knapsack problem which are identical except that each has a different amount of resource, c.

It is easy to construct an algorithm which takes advantage of this fact, but several other points deserve to be mentioned first. The above theorem illuminates the special conditions required for GLM to guarantee an optimum allocation to the desired knapsack problem, namely that the amount of available resource, c, is the sum of the constraint coefficients of the first k most efficient variables. Thus a sufficient condition for optimality with respect to a particular right-hand-side, c, is that $c=C_1+\ldots+C_k$.

[1/] Actually this "proof" neglects the special case of ties in the efficiencies of variables. However, if two or more variables have equal efficiencies, their coefficients in the Lagrangian will have the same sign simultaneously. Thus for any value of $\lambda\geq 0$ the Lagrangian is maximized by making the first k variables equal to one, because k will be such that variables with equal efficiencies will either be all set to 1 or to 0.

While the above-stated sufficient condition for optimality is easily deduced from Theorem 2, the question of necessary conditions is more troublesome. For example, it is easy to construct a counterexample to the assertion that $H(x^*)$ is optimal over the range $C(x^*) \leq c \leq C(x^*)+C'$, where $C' = \min_{k+1 \leq i \leq n} (C_i)$. Fortunately, the computation of an upper bound on the optimal value of the objective function is straightforward. Such an upper bound is the optimal solution to (1) with variables continuous on [0,1], but that is just the linear programming solution (2). If the k most efficient variables are set to their upper bound, then the LP solution sets

$$x_{k+1} = (c - \sum_{i=1}^{k} C_i)/C_{k+1}$$

so that the optimum value of the objective function for the continuous problem is:

$$\sum_{i=1}^{k} H_i + [H_{k+1}(c - \sum_{i=1}^{k} C_i)]/C_{k+1}$$

and thus the upper bound on the optimum objective function value for the binary-variable problem is:

$$H(x^*) + (H_{k+1}/C_{k+1})\hat{c},$$

where $\hat{c}$ is the difference between the right-hand-side, c, and the value $C(x^*)$ generated by the GLM solution x^*, and where

$$H(x^*) = \sum_{i=1}^{k} H_i, \quad C(x^*) = \sum_{i=1}^{k} C_i.$$

The reason an upper bound is desirable, of course, is that x^* may not be optimal if $c \neq C(x^*)$. "Gaps" are said to occur when the desired value of c is produced by none of the solutions maximizing the Lagrangian resulting from (1). For knapsack problems, GLM gaps are particularly easy to visualize; they may be thought of as the jumps in the step function of $C(x^*)$ vs. λ Thus GLM can not guarantee optimality if the desired value of c lies in such a discontinuity. Everett noted that there is no completely satisfactory way to treat such gaps. One of the advantages of computing critical multiplier values rather than searching for different allocations is that a special type of gap can be easily filled. Suppose the jth and j=1th variables have equal efficiencies; as already noted, no value of λ

exists which will simultaneously make the coefficient $H_j - \lambda C_j$ positive, and $H_{j+1} - \lambda C_{j+1}$ negative, so that GLM will find the optimum allocations $x_1 = \ldots = x_j = 1$, and $x_1 = \ldots = x_{j+1} = 1$, but not the optimum allocation $x_1 = \ldots = x_{j-1} = x_{j+1} = 1$, $x_j = 0$, although this is an optimum allocation for $c = C(x^*) = C_1 + \ldots + C_{j-1} + C_{j+1}$.

The above type of gap is easily remedied, and although gaps can still cause trouble, many gaps can be handled by "slipping" a 1 from one location in x^* to another. The method is analogous to Everett's use of different multipliers for different cells, and forms the backbone of the Slippery Algorithm.

The Slippery Algorithm

The Slippery Algorithm, a GLM-like technique for solving (1), was born after the optimum-allocation structure of a number of knapsack problems had been closely examined. These examinations prompted two observations, namely that (a) the greater the relative efficiency of a variable, the more likely that the variable would be at its upper bound in the optimum solution (as well as in x^*); and (b) all problems examined could be solved optimally without examining (either explicitly or implicitly) combinations of variables.

The Slippery Algorithm starts by setting the first k most efficient variables equal to one (i.e., finding x^*), then it tries to improve the solution by comparing, one-to-one each variable at its lower bound with each variable at its upper bound in order to determine if a less-efficient variable with a greater payoff might replace a more-efficient variable with a lesser payoff, for a net increase in payoff. The algorithm is given in more detail in Display 1.

Unlike many integer programming problems, it is usually computationally feasible to solve knapsack problems with implicit enumeration methods (e.g., dynamic programming, Balas codes.) Why, then, develop an algorithm which can not guarantee optimality? Firstly, one is often principally interested in sensitivity analysis, and therefore the sequence $[C(x^*)]_{k=1}^{n}$ and the corresponding values of $H(x^*)$ are of

The Slippery Algorithm

(Selection of projects competing for a single resource.)

$$\text{Maximize } H_1x_1+\ldots+H_nx_n$$

$$\text{Subject to } C_1x_1+\ldots+C_nx_n \leq c$$

$$H_i, C_i,\ c > 0;\ x_i=0(\text{reject}) \text{ or } 1\ (\text{accept})$$

STEP 1: Compute the efficiencies, $H_i/C_i=E_i$

STEP 2: Reorder the variables in terms of decreasing efficiency. Let $i=1$.

STEP 3: Define the remainder, $R=c-(C_1+\ldots+C_i)$. Compute R; if $R < 0$, go to step 4; otherwise, let $H(x)=H_1+\ldots+H_i$, and let $x_i=1$. Increase i by 1 and repeat step 3.

STEP 4: Compute the upper bound, $U=H(x)+(E_i)(R)$. Reset $R_{new}=R_{old}+C_i$, increase i by 1.

STEP 5: Check if $C_i < R$. If not, go to step 6. If so, let $R_{new}=R_{old}-C_i$; let $x_i=1$; go to step 6.

STEP 6: If $i < n$, increment i by 1, repeat step 5. If $i=n$, reset $i=1$ and go to step 7.

STEP 7: (a) If $x_i=1$, increment i by one and then either repeat step 7a if $i \leq n$, or go to step 8 if $i > n$.

(b) If $x_i=0$, set $j=1$, then: If $x_j=1$ <u>and</u> $H_j < H_i$ <u>and</u> $C_j \geq R + C_i$ go to step 7c; otherwise, increment j by 1; if $j=i$, go to 7a, otherwise repeat 7b.

(c) $x_j=0$, $x_i=1$, $R_{new}=R_{old}=C_i-C_j$. Go to step 7a.

STEP 8: Stop. Do these projects with variables equal to one. Payoff is $H_1x_1+\ldots+H_nx_n$; unused resource is R. Upper bound on the optimum payoff is U.

- Display 1 -

more value than the solution to the problem with a specific right-hand-side. Secondly, in industrial situations the data is known only approximately; in such cases an "optimum" answer may be no better than a "suboptimum" answer. Thirdly, in many instances one could conceivably be without a digital computer; the Slippery Algorithm is amenable to solution by hand. Fourthly, the author feels the Slippery Algorithm is both easy to understand and intuitively appealing; in the author's experience it is easy for a nonmathematically-oriented manager to use, and the preeminence of the variable efficiencies contribute to managerial insights as no implicit enumeration method could (even assumming a manager could understand such a method).[2/] And finally, the Slippery Algorithm furnishes an upper bound on the optimum solution, so that the user may decide for himself if the difference between the bound and the algorithm's allocation is sufficiently high to justify the slower optimum-seeking methods (and he will have an excellent starting solution).

[2/] Based on experience with managers in the Larimer County, Colorado government, and elsewhere.

COMPUTATIONAL RESULTS

In order to ascertain computational characteristics of the Slippery Algorithm, the author has solved 100 knapsack problems using a FORTRAN IV program of the method on the Digital Equipment Corporation PDP-10 owned by the Colorado School of Mines. Timings were made by a machine-language clock subroutine, and do not include input/output or compilation times. On several problems which were run repeatedly, computational times fluctuated by as much as 35 per cent, probably reflecting variations in the timesharing load.

The problems solved can be broken down as follows: a 10-variable problem with 9 different values of c; a 40-variable problem with 7 different values of c; a 75-variable problem with 22 different values of c; four 100-variable problems each with 10 different values of c; a 150-variable problem with 12 different values of c; and a 200-variable problem with 10 different values of c.

The 10-variable problem is from Trauth and Woolsey (1969), and is an application of an actual situation. The 40-variable problem is from Plateau and Fayard (1971), and was generated from a random-number table. The coefficients of the 200-variable problem (problem set 1) were likewise taken from a random-number table. Two of the four 100-variable problems are merely the first-and last-hundred variables of the 200-variable problem (problem sets 2 and 3 respectively). The other two 100-variable problems reflect more "structured" situations; in one case the efficiencies are all between 0.5 and 1.5 (problem set 4), and in the other case, the constraint coefficients are all between

30 and 85 (problem set 5). The 150-variable (problem set 6) and 75-variable (problem set 7) problems both contain random coefficients, but both also contain groups of variables which "dominate" the problem (by having very high efficiencies and/or very large objective and constraint coefficients). It is felt that this smattering of various types of structured problems together with the more traditional randomly-generated problems give a fuller representation of problem-types which are likely to be encountered in commercial and governmental applications. The output from the various problems and a listing of the program are given in the appendix.

Certain aspects of the computational results are summarized in Table 1 and Figures 1 and 2. The computational experience reported by Plateau and Fayard suggests that the fastest optimum-seeking method for the knapsack problem is the method of Greenberg and Hegerich (1970).[3/] The Slippery Algorithm executed the Trauth and Woolsey problem set in an unknown amount of time[4/]; Plateau and Fayard reported times from 90 to 190 milliseconds for problems from this set when using the Greenberg and Hegerich method on a Honeywell Gamma M 40 computer. The Slippery Algorithm solved all problems in the 40-variable problem set in a total of 2.586 seconds; Plateau and Fayard report a total time of 7.81 seconds (about three times as long) for this problem set when using the Greenberg and Hegerich method. The Slippery Algorithm achieved optimality in all cases for the Trauth and Woolsey problem set, and in three of seven cases for the 40-variable problem set (in the worst case, 96.9 per cent of optimality was achieved). No comparison of run times

---

[3/] The Greenberg and Hegerich method is a branch-search algorithm, which takes advantage of the facts that the linear programming solution to (1) is (2) and that upper- and lower-bounding variables for a binary problem simply requires the removal of columns from the coefficient matrix. The method starts by reordering the variables by decreasing efficiency, and could be considered a modified Slippery Algorithm where the reordering and initial solution portions are retained, but an implicit enumeration technique replaces the less powerful but faster "slipping" feature.

[4/] The clock subroutine is advanced in 17-millisecond increments; variable reordering, inital-solution determination, and slipping were each done between advancements of the clock time.

Problem Set:	Right-Hand Side:	Payoff:	Percentage Of LP Pay-off:*	Percentage Of Variables At Upper-Bound:	Solution Time:** (millisec)
Trauth and Woolsey	55	50	100.0	70.0	0
	60	52	100.0	70.0	0
	65	57	100.0	70.0	0
	70	62	100.0	70.0	0
	75	67	100.0	80.0	0
	80	68	100.0	80.0	0
	85	70	100.0	80.0	0
	90	75	100.0	80.0	0
	100	85	100.0	90.0	0
Plateau and Fayard	999	4136	98.7	62.5	367
	2000	4653	98.0	65.0	366
	3000	4983	100.0	72.5	383
	5000	5164	96.9	77.5	367
	7000	5826	100.0	77.5	367
	8000	5894	99.9	80.0	367
	12,000	6081	100.0	77.5	367
One	500	2127	98.8	15.5	2833
	1500	3767	99.9	26.5	2900
	3000	5582	99.9	38.5	2933
	4000	6581	100.0	45.0	2917
	5000	7483	100.0	50.5	2917
	6000	8304	99.9	57.0	2850
	7000	9016	100.0	63.0	2817
	8000	9655	100.0	69.0	2733
	10,000	10,595	99.9	82.0	2533
	12,000	11,112	100.0	92.5	2317
Two	200	1148	98.5	18.0	1166
	500	1701	99.5	27.0	1182
	1000	2403	99.5	37.0	1216
	1500	2984	99.9	44.0	1199
	2000	3452	99.7	52.0	1183
	2500	3858	99.8	60.0	1183
	3000	4202	99.8	70.0	1182
	3500	4488	100.0	76.0	1082
	4000	4673	100.0	84.0	1083
	4500	4804	100.0	92.0	1016

*Percentage of integer optimum for the Trauth and Woolsey and Plateau and Fayard problem sets.
**To within 17 milliseconds.

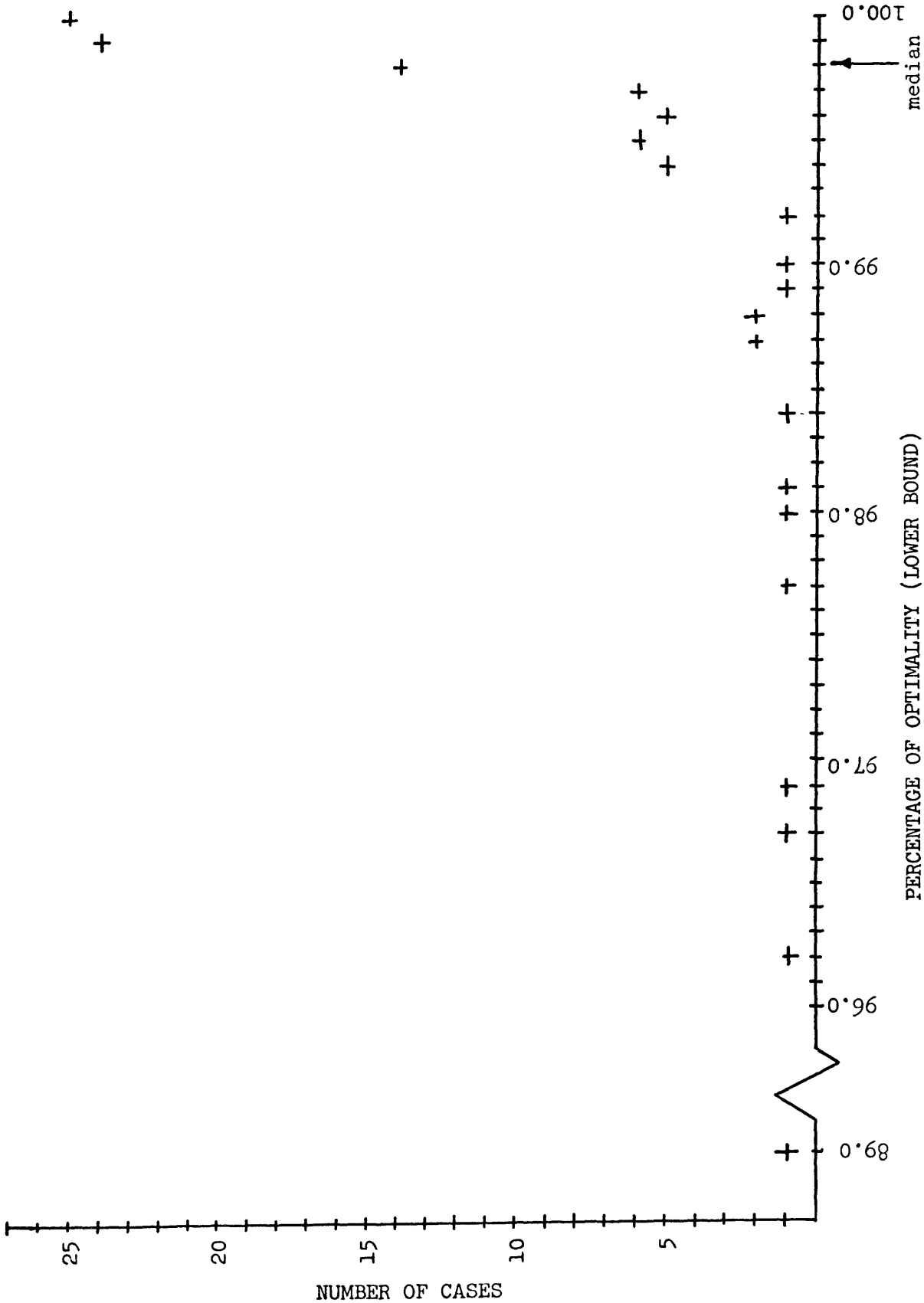
-TABLE 1a-

Problem Set:	Right-Hand-Side:	Payoff:	Percentage Of LP Pay-off:	Percentage Of Variables At Upper-Bound:	Solution Time: (millisec)
Three	500	1319	99.4	18.0	1333
	1000	2016	99.7	27.0	1350
	1500	2593	99.8	35.0	1400
	2000	3091	99.5	41.0	1366
	2500	3561	99.5	45.0	1350
	3000	4008	99.6	48.0	1350
	3500	4420	99.5	54.0	1350
	4000	4799	99.9	58.0	1393
	5000	5430	99.9	68.0	1300
	6000	5910	99.8	79.0	1267
Four	500	719	99.4	10.0	1167
	1000	1359	99.0	19.0	1183
	1500	2957	99.8	27.0	1199
	2000	2494	99.8	34.0	1199
	2500	2995	99.8	41.0	1216
	3000	3468	99.9	46.0	1199
	3500	3920	99.8	51.0	1183
	4000	4358	99.9	55.0	1183
	5000	5163	99.9	69.0	1150
	6000	5854	99.9	81.0	1083
Five	200	438	89.0	5.0	1383
	500	978	98.1	12.0	1383
	1000	1708	99.9	22.0	1417
	1500	2340	99.8	31.0	1400
	2000	2900	99.8	39.0	1416
	2500	3380	99.9	46.0	1433
	3000	3794	99.8	54.0	1416
	3500	4148	99.9	61.0	1383
	4000	4439	99.9	70.0	1333
	4500	4659	99.9	78.0	1350
Six	500	4264	100.0	14.0	2067
	1000	5065	99.8	20.0	2133
	1500	5763	99.4	27.3	2133
	2000	6457	99.6	29.3	2133
	2500	7102	99.6	32.0	2234
	3000	7706	100.0	37.3	2150
	3500	8192	99.7	44.0	2367
	4000	8641	99.1	48.0	2150
	5000	9666	99.8	47.3	2134
	6000	10,480	100.0	55.3	2134

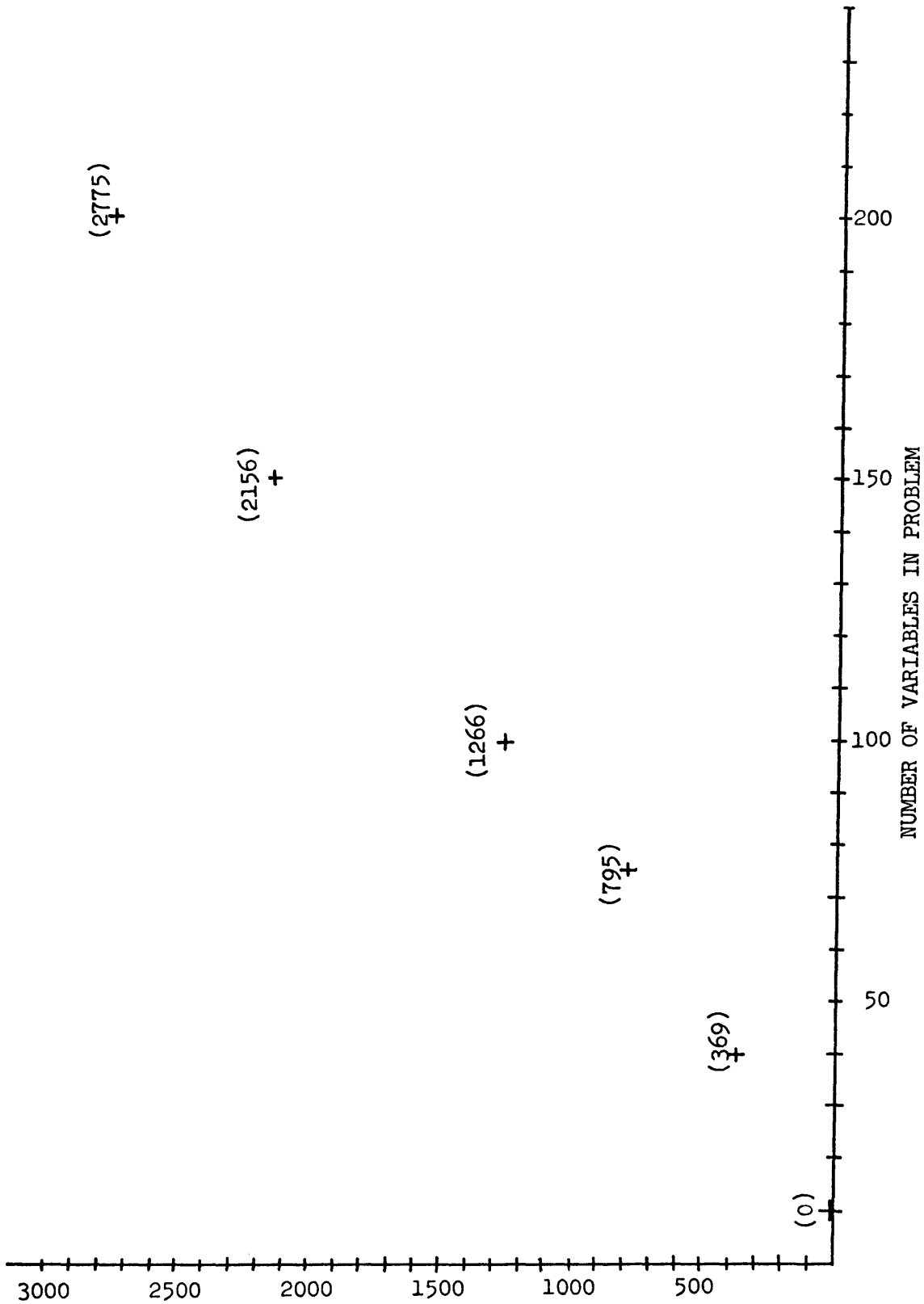
-TABLE 1b-

Problem Set:	Right-Hand-Side:	Payoff:	Percentage Of LP Pay-off:	Percentage Of Variables At Upper-Bound:	Solution Time: (millisec)
Six	7000	11,185	100.0	60.0	2117
	8000	11,709	99.0	71.3	2001
Seven	1500	5696	99.6	37.3	816
	2000	6334	99.7	44.0	816
	2500	6794	97.7	41.3	817
	3000	7487	99.7	41.3	800
	3100	7585	99.7	42.7	817
	3200	7646	99.2	45.3	800
	3300	7770	99.5	42.7	800
	3400	7859	99.4	49.3	816
	3500	7909	98.7	54.7	816
	4000	8414	98.9	53.3	800
	4500	8970	99.9	52.0	817
	4600	9059	99.9	53.3	800
	4700	9109	99.4	56.0	800
	4800	9242	99.8	52.0	800
	4900	9340	99.9	53.3	800
	5000	9401	99.6	56.0	817
	6000	10,225	99.9	62.7	800
	6500	10,506	99.1	70.7	766
	8000	11,556	99.9	72.0	784
	10,000	12,024	98.8	85.3	750
	12,000	12,204	96.7	89.3	734
	14,000	12,878	99.9	88.0	733

-TABLE 1c-


100.0
median
99.6
98.6
97.6
96.6
89.6
PERCENTAGE OF OPTIMALITY (LOWER BOUND)
25
20
15
10
5
NUMBER OF CASES

-FIGURE 1-



MEAN EXECUTION TIME (MILLISECONDS)

-FIGURE 2-

has been made for the other problem sets, and the optimum integer solution has not been determined (the linear programming solution being considered as an adequate, though often high, measure of the integer-variable optimum).

Values of c were such that the percentage of variables finally set to their upper bound varied from 5 per cent to 92.5 per cent. In all cases, the final allocation was between 89.0 and 100.0 per cent of the LP objective function value (of the integer-variable optimum in the 10- and 40-variable problem sets). Of the 100 problems solved, in 96 cases the final allocation was not less than 98.0 per cent of the LP payoff. Execution time appears to have a roughly linear relationship to the number of variables. A number of investigators have reported that such is not often the case with implicit-enumeration techniques, where in fact computational time usually increases very rapidly with problem size. For each problem set, execution times were highest when approximately 30 to 60 per cent of the variables achieved their upper bound, and were lowest when the largest portion of the variables was upper-bounded.

As both problem size and the percentage of variables set to 1 increases, the percentage of the LP optimum achieved increases. This is to be expected, as in the former case there is a tendency to fill the holes between efficiencies, and as in the latter case the difference between initial solution and LP solution becomes less critical since the variable given a fractional value in the LP solution (x_r in (2)) has a relatively small efficiency.

CONCLUSIONS

Several conclusions suggest themselves as a result of the investigations reported in this work:

1. For one-constraint binary programming problems, GLM is not a good solution approach, as the Slippery Algorithm is considerably quicker (since one need not search for the multiplier values) and the "slipping" feature is much less complicated than the simultaneous use of different multiplier values.
2. Based on the computation reported in this work, as the size of the knapsack problem increases, the optimal integer-variable solution becomes increasingly close to the optimal continuous-variable solution.
3. Based on the computation reported in this work, the Slippery Algorithm determines an allocation so nearly optimal that there seems to be little justification for using an implicit-enumeration method. This is especially true for large problems, as both the goodness of the Slippery Algorithm's allocation and the relative efficiency of the algorithm increase with the size of the problem.

While conclusions based on 100 problem tests might be thought of as tenuous, the consistency with which these tests support the above conclusions and the wide variation in problem types together suggest that there is little chance that further tests will overturn the above conclusions.

The author would recommend the Slippery Algorithm whenever (1) the coefficients are subject to a degree of uncertainty, or (2) the economic impact of using a possibly suboptimal, but nonetheless very

good allocation is overshadowed by estimated increases in computing expenses necessary for implicit-enumeration methods, or (3) the computation must be done by hand, or (4) a simple-to-understand solution method would enhance the chances of implementing a proposed allocation. The author would also recommend that the Slippery Algorithm be used to determine a lower bound on the optimal allocation, in the case that an investigator felt warranted in using an implicit-enumeration technique.

APPENDIX

Contained in the following pages is the listing of the FORTRAN IV program used to solve the knapsack problems reported in this work. Also contained is the output from the 100 knapsack problems solved.

```
C       THE SLIPPERY ALGORITHM
C            HEURISTIC FOR 0-1 KNAPSACK PROBLEMS
C
        INTEGER X(250)
        DIMENSION A(250),C(250),NEXT(250),RATIO(251),ITLE(14)
        WRITE(5,100)
100     FORMAT( INPUT NUMBERS FOR INPUT, OUTPUT FILES '/)
        READ(5,102)IO1,IO2
102     FORMAT(2I)
        READ(IO1,300)(ITLE(I) I=1,14)
300     FORMAT(14A5)
        READ(IO1,101)N
101     FORMAT(I3)
        READ(IO1,103) (C(I),A(I),I=1,N)
103     FORMAT(2F6.2)
        READ(IO1,104)RHS
104     FORMAT(F6.2)
        WRITE(IO2,301)(ITLE(I),I=1,14)
301     FORMAT(30X,26H***THE SLIPPERY ALGORITHM***///20X,
        1 14A5///10X,8HVARIABLE,15X,12HCOEFFICIENTS,15X,
        2 5HRATIO/29X,22HOBJECTIVE   CONSTRAINT/)
302     FORMAT(12X,I3,13X,F8.1,5X,F8.1,8X,F10.5)
        CALL RTIME(M1)
        DO 2 I=1,N
2       RATIO(I) = C(I)/A(I)
        WRITE(IO2,302)(I,C(I),A(I),RATIO(I),I=1,N)
        MAX=1
        MIN=1
        DO 60 K=2,N
        IF(RATIO(MAX).GT.RATIO(K)) GO TO 220
210     NEXT(K)=MAX
        MAX=K
        GO TO 60
220     IF(RATIO(K).GT.RATIO(MIN)) GO TO 240
        NEXT(MIN)=K
        MIN=K
        GO TO 60
240     I=MAX
245     LAST=I
        I=NEXT(I)
        IF(RATIO(I).GT.RATIO(K)) GO TO 245
        NEXT(LAST)=K
        NEXT(K)=I
60      CONTINUE
        CALL RTIME(M2)
C*****THE ABOVE STEPS REORDER THE X(I)'S BY DECREASING
C*****BANG-FOR-BUCK.
20      ZCUM = 0.0
        REMAIN = RHS
        I1=MAX
21      IF(I1.EQ.MIN) GO TO 999
        IF(REMAIN.GE.A(I1)) GO TO 4
        IEP=IEP+1
        IF(IEP.EQ.1)GO TO 15
16      I1=NEXT(I1)
        GO TO 21
15      BOUND=REMAIN*RATIO(I1)+ZCUM
        GO TO 16
4       X(I1)=1
        ZCUM=ZCUM+C(I1)
```

```
      REMAIN=REMAIN-A(I1)
      I1=NEXT(I1)
      GO TO 21
C***GENERATES THE STARTING FEASIBLE SOLUTION
999   I=MAX
      CALL RTIME(M3)
1004  IF(I.EQ.MIN) GO TO 1
      IF(X(I).EQ.1) GO TO 1002
      J1=MAX
1003  IF(J1.EQ.I) GO TO 1002
      IF(X(J1).EQ.0) GO TO 1001
      SUM = REMAIN + A(J1)
      IF(C(J1).GE.C(I).OR.SUM.LT.A(I))GO TO 1001
      X(J1)=0
      X(I)=1
      REMAIN=SUM-A(I)
      ZCUM=ZCUM-C(J1)+C(I)
      GO TO 1002
1001  J1=NEXT(J1)
      GO TO 1003
1002  I=NEXT(I)
      GO TO 1004
10    CALL RTIME(M4)
      WRITE(IO2,303)RHS,BOUND,ZCUM,REMAIN
303   FORMAT(///10X,17HRIGHT-HAND-SIDE: ,F8.1/10X,
      124HUPPER-BOUND ON OPTIMUM: ,F8.1/10X,8HPAYOFF: ,F8.1
      2,5X ,16HRESOURCES LEFT: ,F6.1//10X,10HDO THESE: )
      DO 14 I=1,N
      IF(X(I).EQ.1) WRITE(IO2,304) I
304   FORMAT(13X,I3)
14    X(I)=0
C     RESETS THE VARIABLES FOR ANOTHER RUN
      M1=M2-M1
      M2=M3-M2
      M3=M4-M3
      WRITE(IO2,305)M1,M2,M3
305   FORMAT(//10X,29HTIMING SUMMARY, MILLISECONDS:/10X,
      1 22HORDERING VARIABLES: ,I5/10X,18HINITIAL SOLUTION: ,
      2I5/10X,10HSLIPPING: ,I5)
      READ(IO1,104)RHS
      IF(RHS.LE.0.0) STOP
      CALL RTIME(M2)
      M1=M2
      IEP=0
      GO TO 20
      END
```

THE SLIPPERY ALGORITHM

THE TRAUTH & WOOLSEY PROBLEM SET

VARIABLE	COEFFICIENTS OBJECTIVE	COEFFICIENTS CONSTRAINT	RATIO
1	20.0	30.0	0.00000
2	18.0	25.0	0.00000
3	17.0	20.0	0.00000
4	15.0	18.0	0.00000
5	15.0	17.0	0.00000
6	10.0	11.0	0.00000
7	5.0	5.0	0.00000
8	3.0	2.0	0.00000
9	1.0	1.0	0.00000
10	1.0	1.0	0.00000

RIGHT-HAND-SIDE: 55.0
UPPER-BOUND ON OPTIMUM: 50.3
PAYOFF: 50.0 RESOURCES LEFT: 0.0

DO THESE:
4
5
6
7
8
9
10

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 0
SLIPPING: 0

RIGHT-HAND-SIDE: 60.0
UPPER-BOUND ON OPTIMUM: 54.5
PAYOFF: 52.0 RESOURCES LEFT: 3.0

DO THESE:
3
5
6
7
8
9
10

TIMING SUMMARY MILLISECONDS:
ORDERING VARIABLES: 0

```
INITIAL SOLUTION:      0
SLIPPING:      0


RIGHT-HAND-SIDE:    65.0
UPPER-BOUND ON OPTIMUM:    58.7
PAYOFF:    57.0      RESOURCES LEFT:     1.0

DO THESE:
      3
      4
      5
      7
      8
      9
     10


TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:      0


RIGHT-HAND-SIDE:    70.0
UPPER-BOUND ON OPTIMUM:    62.8
PAYOFF:    62.0      RESOURCES LEFT:     0.0

DO THESE:
      3
      4
      5
      6
      8
      9
     10


TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:      0


RIGHT-HAND-SIDE:    75.0
UPPER-BOUND ON OPTIMUM:    67.0
PAYOFF:    67.0      RESOURCES LEFT:     0.0

DO THESE:
      3
      4
      5
      6
      7
      8
      9
     10
```

```
TIMING SUMMARY  MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:      0


RIGHT-HAND-SIDE:    80.0
UPPER-BOUND ON OPTIMUM:    70.6
PAYOFF:    68.0      RESOURCES LEFT:     0.0

DO THESE:
      2
      4
      5
      6
      7
      8
      9
     10


TIMING SUMMARY  MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:      0


RIGHT-HAND-SIDE:    85.0
UPPER-BOUND ON OPTIMUM:    74.2
PAYOFF:    70.0      RESOURCES LEFT:     2.0

DO THESE:
      2
      3
      4
      6
      7
      8
      9
     10


TIMING SUMMARY  MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:      0


RIGHT-HAND-SIDE:    90.0
UPPER-BOUND ON OPTIMUM:    77.8
PAYOFF:    75.0      RESOURCES LEFT:     1.0

DO THESE:
      2
      3
```

```
     4
     5
     7
     8
     9
    10

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:      0


RIGHT-HAND-SIDE:  100.0
UPPER-BOUND ON OPTIMUM:    77.8
PAYOFF:    85.0      RESOURCES LEFT:     0.0

DO THESE:
     2
     3
     4
     5
     6
     7
     8
     9
    10

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:      0
```

*** HI SLIPPERY ALGORITHM ***

THE PLATEAU AND PAYAND PROBLEM SET

VARIABLE	COEFFICIENTS OBJECTIVE	CONSTRAINT	RATIO
1	54.0	541.0	0.9982
2	[illegible]	756.0	0.[illegible]651
3	780.0	3541.0	0.[illegible]2282
4	[illegible].0	75.0	2.[illegible]6667
5	24.0	1.0	24.0000
6	35.0	692.0	0.[illegible]5368
7	61.0	843.0	0.7236
8	42.0	21.0	2.0000
9	785.0	10.0	78.[illegible]0000
10	43.0	4[illegible].0	1.[illegible]7500
11	512.0	46.0	11.3043
12	424.0	51.0	8.[illegible]1373
13	321.0	752.0	0.[illegible]2686
14	11.0	810.0	0.[illegible]1358
15	4.0	510.0	0.[illegible]7784
16	47.0	21.0	2.[illegible]3610
17	35.0	42.0	0.83333
18	56.0	121.0	0.46281
19	12.0	5.0	2.[illegible]000
20	754.0	4.0	88.[illegible]00
21	55.0	72.0	2.6389
22	457.0	631.0	0.[illegible]2425
23	1.0	724.0	0.[illegible]0139
24	65.0	435.0	0.4943
25	74.0	2.0	37.[illegible]000
26	22.0	820.0	0.[illegible]2683
27	4.0	64.0	0.[illegible]252
28	45.0	73.0	0.61644
29	12.0	77[illegible].0	0.41558
30	5.0	43.0	0.[illegible]1628
31	5[illegible].0	85.0	0.57[illegible]59
32	57.0	912.0	0.06250
33	23.0	4.0	5.[illegible]000
34	54.0	35.0	1.[illegible]4286
35	2.0	14.0	0.[illegible]4286
36	4[illegible].0	42.0	1.[illegible]000
37	77.0	22.0	3.[illegible]000
38	76.0	54.0	1.44444
39	682.0	32.0	21.[illegible]250
40	142.0	33.0	4.5714

RIGHT-HAND-SIDE: 99[illegible]
UPPER-BOUND ON OPTIMUM: 4249.8
PAYOFF: 4136.0 RESOURCES LEFT: 49.0

DO THESE:
4
5

8
9
10
11
12
16
17
18
19
20
21
25
26
30
31
33
34
35
36
37
38
39
40

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 400
INITIAL SOLUTION: 0
SLIPPING: 33

RIGHT-HAND-SIDE: 2000.0
UPPER-BOUND ON OPTIMUM: 4789.2
PAYOFF: 4653.0 RESOURCES LEFT: 27.0

DO THESE:
4
5
8
9
10
11
12
16
17
18
19
20
21
22
24
25
28
31
33
34
35
36
37

```
38
39
40

TIMING SUMMARY MILLI SECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      17
SLIPPING:      1


RIGHT-HAND-SIDE: 3000.0
UPPER-BOUND ON OPTIMUM: 5068.3
PAYOFF: 4983.0      RESOURCES LEFT:  168.0

DO THESE:
4
5
8
9
10
11
12
13
16
17
18
19
20
21
22
24
25
27
28
30
31
33
34
35
36
37
38
39
40

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:      17


RIGHT-HAND-SIDE: 5001.0
UPPER-BOUND ON OPTIMUM: 5514.0
PAYOFF: 5164.0      RESOURCES LEFT:  12.0

DO THESE:
```

1
2
4
5
7
8
9
10
11
12
13
16
17
18
19
20
21
22
24
25
27
28
30
31
33
34
36
37
38
39
40

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 0
SLIPPING: 17

RIGHT-HAND-SIDE: 7000.0
UPPER-BOUND ON OPTIMUM: 5835.0
PAYOFF: 5826.0 RESOURCES LEFT: 86.0

DO THESE:

1
3
4
5
8
9
10
11
12
13
16
17
18
19
20

```
      21
      22
      24
      25
      27
      28
      30
      31
      33
      34
      35
      36
      37
      38
      39
      40

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:     16
SLIPPING:     17


RIGHT-HAND-SIDE: 8000.0
UPPER-BOUND ON OPTIMUM: 5916.3
PAYOFF: 5894.0      RESOURCES LEFT:  3[illegible]0.0

DO THESE:
      1
      2
      3
      4
      5
      8
      9
     10
     11
     12
     13
     16
     17
     18
     19
     20
     21
     22
     24
     25
     27
     28
     30
     31
     33
     34
     35
     36
     37
     38
```

39
40

TIMING SUMMARY MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 0
SLIPPING 1

RIGHT-HAND-SIDE: ******
UPPER-BOUND ON OPTIMUM: 6085.1
PAYOFF 6081.0 RESOURCES LEFT: 3.3.0

DO THESE:
1
2
3
4
5
6
7
8
9
10
11
12
13
16
17
18
19
20
21
22
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40

TIMING SUMMARY MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 0
SLIPPING: 1

THE SLIPPERY ALGORITHM

****PROBLEM SET ONE****

VARIABLE	COEFFICIENTS OBJECTIVE	CONSTRAINT	RATIO
1	48.0	68.0	0.70588
2	1.0	73.0	0.01370
3	53.0	95.0	0.55789
4	1.0	93.0	0.01075
5	64.0	95.0	0.67368
6	64.0	98.0	0.65306
7	17.0	82.0	0.20732
8	19.0	2.0	9.50000
9	59.0	65.0	0.90769
10	20.0	95.0	0.21053
11	96.0	66.0	1.45455
12	57.0	74.0	0.77027
13	29.0	15.0	1.93333
14	70.0	5.0	14.00000
15	92.0	62.0	1.48387
16	90.0	5.0	18.00000
17	8.0	20.0	0.40000
18	75.0	94.0	0.79787
19	75.0	72.0	1.04167
20	49.0	16.0	3.06250
21	60.0	94.0	0.63830
22	65.0	38.0	1.71053
23	65.0	13.0	5.00000
24	5.0	40.0	0.12500
25	91.0	13.0	7.00000
26	82.0	69.0	1.18841
27	39.0	34.0	1.14706
28	88.0	52.0	1.69231
29	85.0	68.0	1.25000
30	76.0	82.0	0.92683
31	82.0	4.0	20.50000
32	25.0	68.0	0.36765
33	45.0	68.0	0.66176
34	30.0	85.0	0.35294
35	75.0	22.0	3.40909
36	14.0	19.0	0.73684
37	44.0	88.0	0.50000
38	90.0	55.0	1.63636
39	96.0	33.0	2.90909
40	99.0	19.0	5.21053
41	60.0	44.0	1.36364
42	71.0	57.0	1.24561
43	15.0	57.0	0.26316
44	63.0	76.0	0.82895
45	82.0	60.0	1.36667
46	17.0	26.0	0.65385
47	61.0	66.0	0.92424
48	83.0	35.0	2.37143
49	92.0	87.0	1.05747
50	41.0	30.0	1.36667

51	13.0	60.0	0.21667
52	41.0	51.0	0.80392
53	55.0	65.0	0.84615
54	24.0	10.0	2.40000
55	50.0	49.0	1.02041
56	46.0	6.0	7.66667
57	13.0	21.0	0.61905
58	16.0	3.0	5.33333
59	74.0	30.0	2.46667
60	92.0	3.0	30.66667
61	24.0	96.0	0.25000
62	36.0	89.0	0.40449
63	1.0	41.0	0.02439
64	22.0	17.0	1.29412
65	2.0	27.0	0.07407
66	51.0	63.0	0.80952
67	7.0	41.0	0.17073
68	76.0	49.0	1.55102
69	97.0	82.0	1.18293
70	45.0	24.0	1.87500
71	36.0	70.0	0.51429
72	27.0	66.0	0.40909
73	73.0	99.0	0.73737
74	20.0	72.0	0.27778
75	37.0	1.0	37.00000
76	63.0	42.0	1.50000
77	71.0	11.0	6.45455
78	84.0	20.0	4.20000
79	52.0	54.0	0.96296
80	22.0	36.0	0.61111
81	78.0	70.0	1.11429
82	17.0	23.0	0.73913
83	96.0	65.0	1.47692
84	18.0	59.0	0.30508
85	21.0	21.0	1.00000
86	40.0	57.0	0.70175
87	29.0	2.0	14.50000
88	96.0	89.0	1.07865
89	29.0	17.0	1.70588
90	21.0	37.0	0.56757
91	15.0	47.0	0.31915
92	34.0	42.0	0.80952
93	33.0	97.0	0.34021
94	6.0	48.0	0.12500
95	95.0	68.0	1.39706
96	15.0	43.0	0.34884
97	16.0	33.0	0.48485
98	6.0	21.0	0.28571
99	11.0	30.0	0.36667
100	70.0	81.0	0.86420
101	51.0	42.0	1.21429
102	38.0	58.0	0.65517
103	19.0	21.0	0.90476
104	66.0	30.0	2.20000
105	71.0	32.0	2.21875
106	5.0	86.0	0.05814
107	13.0	5.0	2.60000
108	78.0	7.0	11.14286
109	84.0	54.0	1.55556
110	66.0	58.0	1.13793

111	93.0	74.0	1.25676
112	59.0	46.0	1.28261
113	37.0	63.0	0.58730
114	30.0	64.0	0.46875
115	6.0	46.0	0.13043
116	54.0	9.0	6.00000
117	4.0	44.0	0.09091
118	53.0	78.0	0.67949
119	62.0	83.0	0.74699
120	95.0	42.0	2.26190
121	78.0	92.0	0.84783
122	11.0	83.0	0.13253
123	81.0	80.0	1.01250
124	50.0	35.0	1.42857
125	34.0	81.0	0.41975
126	52.0	54.0	0.96296
127	81.0	150.0	0.54000
128	11.0	155.0	0.07097
129	50.0	254.0	0.19685
130	36.0	167.0	0.21557
131	61.0	117.0	0.52137
132	35.0	178.0	0.19663
133	25.0	216.0	0.11574
134	98.0	274.0	0.35766
135	93.0	163.0	0.57055
136	78.0	111.0	0.70270
137	10.0	283.0	0.03534
138	99.0	114.0	0.86842
139	10.0	103.0	0.09709
140	90.0	126.0	0.71429
141	165.0	254.0	0.64961
142	140.0	172.0	0.81395
143	151.0	140.0	1.07857
144	169.0	188.0	0.89894
145	158.0	165.0	0.95758
146	129.0	138.0	0.93478
147	128.0	142.0	0.90141
148	222.0	120.0	1.85000
149	156.0	165.0	0.94545
150	144.0	123.0	1.17073
151	183.0	101.0	1.81188
152	150.0	176.0	0.85227
153	174.0	117.0	1.48718
154	107.0	204.0	0.52451
155	8.0	87.0	0.09195
156	59.0	56.0	1.05357
157	60.0	42.0	1.42857
158	46.0	69.0	0.66667
159	64.0	11.0	5.81818
160	22.0	46.0	0.47826
161	16.0	31.0	0.51613
162	35.0	6.0	5.83333
163	85.0	100.0	0.85000
164	98.0	75.0	1.30667
165	32.0	88.0	0.36364
166	52.0	90.0	0.57778
167	16.0	15.0	1.06667
168	62.0	98.0	0.63265
169	82.0	32.0	2.56250
170	38.0	80.0	0.47500

171	73.0	17.0	4.29412
172	32.0	23.0	1.39130
173	41.0	61.0	0.67213
174	44.0	37.0	1.18919
175	96.0	73.0	1.31507
176	20.0	6.0	3.33333
177	42.0	59.0	0.71186
178	78.0	85.0	0.91765
179	5.0	100.0	0.05000
180	22.0	64.0	0.34375
181	24.0	69.0	0.34783
182	54.0	24.0	2.25000
183	35.0	83.0	0.42169
184	19.0	87.0	0.21839
185	11.0	52.0	0.21154
186	91.0	91.0	1.00000
187	60.0	83.0	0.72289
188	19.0	46.0	0.41304
189	38.0	35.0	1.08571
190	64.0	2.0	32.00000
191	14.0	49.0	0.28571
192	82.0	74.0	1.10811
193	66.0	23.0	2.86957
194	44.0	33.0	1.33333
195	100.0	97.0	1.03093
196	35.0	52.0	0.67308
197	35.0	70.0	0.50000
198	19.0	24.0	0.79167
199	63.0	18.0	3.50000
200	29.0	86.0	0.33721

RIGHT-HAND-SIDE: 500.0
UPPER-BOUND ON OPTIMUM: 2152.6
PAYOFF: 2127.0 RESOURCES LEFT: 1.0

DO THESE:

8
14
16
20
23
25
31
35
39
40
48
54
56
58
59
60
75
77
78
87
104
105

```
107
108
116
120
159
162
169
171
176
182
190
193

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:  210
INITIAL SOLUTION:    17
SLIPPING:   716

RIGHT-HAND-SIDE:   1500.0
UPPER-BOUND ON OPTIMUM:   3772.5
PAYOFF:   3767.0      RESOURCES LEFT:    0.0

DO THESE:
  6
 11
 13
 14
 16
 20
 22
 23
 25
 28
 31
 35
 38
 39
 40
 48
 54
 56
 58
 59
 60
 68
 70
 75
 76
 77
 78
 83
 87
 89
 95
104
105
107
```

```
10P
109
116
120
124
148
151
153
157
159
162
169
171
172
176
182
190
193
199

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:     17
SLIPPING:    783

RIGHT-HAND-SIDE:    3000.0
UPPER-BOUND ON OPTIMUM:    5586.5
PAYOFF:    5582.0      RESOURCES LEFT:     0.0

DO THESE:
  8
 11
 13
 14
 15
 16
 20
 22
 23
 25
 26
 27
 28
 29
 31
 35
 38
 39
 40
 41
 42
 45
 48
 50
 54
 56
 58
```

59
60
64
68
69
70
75
76
77
78
81
83
87
89
95
101
104
105
107
108
109
110
111
112
116
120
124
143
148
150
151
153
157
159
162
164
169
171
172
174
175
176
182
189
190
192
193
194
195
199

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 0
SLIPPING: 833

RIGHT-HAND-SIDE: 4000.0

UPPER-BOUND ON OPTIMUM: 6582.9
PAYOFF: 6581.0 RESOURCES LEFT: 2.0

DO THESE:

8
11
13
14
15
16
19
20
22
23
25
26
27
28
29
31
35
38
39
40
41
42
45
48
49
50
54
55
56
58
59
60
64
68
69
70
75
76
77
78
79
81
83
85
87
88
89
95
101
104
105
107
108
109
110
111

```
112
116
120
123
124
126
143
145
148
149
150
151
153
156
157
159
162
164
167
169
171
172
174
175
176
182
186
189
190
192
193
194
195
199

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:    0
INITIAL SOLUTION:    17
SLIPPING:    800

RIGHT-HAND-SIDE:    5000.0
UPPER-BOUND ON OPTIMUM:    7486.0
PAYOFF:    7483.0      RESOURCES LEFT:    1.0

DO THESE:
    8
    9
   11
   13
   14
   15
   16
   19
   20
   22
   23
   25
```

26
27
28
29
30
31
35
36
38
39
40
41
42
45
47
48
49
50
54
55
56
58
59
60
64
68
69
70
75
76
77
78
79
81
83
85
87
88
89
95
100
101
103
104
105
107
108
109
110
111
112
116
120
123
124
126
138
143
144
145

146
147
148
149
150
151
153
156
157
159
162
164
167
169
171
172
174
175
176
178
182
186
189
190
192
193
194
195
199

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 17
SLIPPING: 800

RIGHT-HAND-SIDE: 6000.0
UPPER-BOUND ON OPTIMUM: 8310.1
PAYOFF: 8304.0 RESOURCES LEFT: 2.0

DO THESE:
8
9
11
13
14
15
16
18
19
20
22
23
25
26
27
28
29

30
31
35
36
38
39
40
41
42
44
45
47
48
49
50
52
53
54
55
56
57
58
59
60
64
66
68
69
70
75
76
77
78
79
81
82
83
85
87
88
89
92
95
100
101
103
104
105
107
108
109
110
111
112
116
120
121
123
124
126

```
138
142
143
144
145
146
147
148
149
150
151
152
153
156
157
159
162
163
164
167
169
171
172
174
175
176
178
182
186
189
190
192
193
194
195
198
199

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:     17
SLIPPING:    733

RIGHT-HAND-SIDE:    7000.0
UPPER-BOUND ON OPTIMUM:    9018.5
PAYOFF:    9016.2      RESOURCES LEFT:     0.0

DO THESE:
    1
    5
    8
    9
   11
   12
   13
   14
   15
```

16
18
19
20
22
23
25
26
27
28
29
30
31
35
36
38
39
40
41
42
44
45
46
47
48
49
50
52
53
54
55
56
58
59
60
64
66
68
69
70
73
75
76
77
78
79
81
82
83
85
87
88
89
92
95
100
101
103
104
105

107
108
109
110
111
112
116
118
119
120
121
123
124
126
136
138
140
142
143
144
145
146
147
148
149
150
151
152
153
156
157
158
159
162
163
164
167
169
171
172
174
175
176
177
178
182
186
187
189
190
192
193
194
195
196
198
199

TIMING SUMMARY, MILLISECONDS:

ORDERING VARIABLES: 0
INITIAL SOLUTION: 0
SLIPPING: 717

RIGHT-HAND-SIDE: 6000.0
UPPER-BOUND ON OPTIMUM: 9656.1
PAYOFF: 9655.4 RESOURCES LEFT: 2.0

DO THESE:

1
5
6
8
9
11
12
13
14
15
16
18
19
20
21
22
23
25
26
27
28
29
30
31
33
35
36
38
39
40
41
42
44
45
46
47
48
49
50
52
53
54
55
56
57
58
59
60
64

66
68
69
70
73
75
76
77
78
79
80
81
82
83
85
86
87
88
89
92
95
100
101
102
103
104
105
107
108
109
110
111
112
113
116
118
119
120
121
123
124
126
136
138
140
141
142
143
144
145
146
147
148
149
150
151
152
153
156
157

158
159
162
163
164
166
167
168
169
171
172
173
174
175
176
177
178
182
186
187
189
190
192
193
194
195
196
198
199

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 0
SLIPPING: 633

RIGHT-HAND-SIDE: 10000.2
UPPER-BOUND ON OPTIMUM: 10023.5
PAYOFF: 12595.0 RESOURCES LEFT: 8.0

DO THESE:
1
3
5
6
8
9
11
12
13
14
15
16
17
18
19
20
21

22
23
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
44
45
46
47
48
49
50
52
53
54
55
56
57
58
59
60
62
64
66
68
69
70
71
72
73
75
76
77
78
79
80
81
82
83
85
86
87
88
89
90

91
92
95
96
97
99
100
101
102
103
104
105
107
108
109
110
111
112
113
114
116
118
119
120
121
123
124
125
126
127
131
135
136
138
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
156
157
158
159
160
161
162
163
164
165
166

```
167
168
169
170
171
172
173
174
175
176
177
178
182
183
186
187
188
189
190
192
193
194
195
196
197
198
199

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:     16
SLIPPING:   417

RIGHT-HAND-SIDE:  12000.0
UPPER-BOUND ON OPTIMUM:  11112.4
PAYOFF:  11112.0      RESOURCES LEFT:     2.0

DO THESE:
    1
    3
    5
    6
    7
    8
    9
   10
   11
   12
   13
   14
   15
   16
   17
   18
   19
   20
   21
```

22
23
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
64
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84

85
86
87
88
89
90
91
92
93
95
96
97
98
99
100
101
102
103
104
105
107
108
109
110
111
112
113
114
115
116
118
119
120
121
123
124
125
126
127
129
130
131
132
134
135
136
138
140
141
142
143
144
145
146
147
148
149
150
151
152

153
154
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 17
SLIPPING: 200

THE SLIPPERY ALGORITHM

PROBLEM SET 2

VARIABLE	COEFFICIENTS OBJECTIVE	CONSTRAINT	RATIO
1	53.0	95.0	0.55789
2	1.0	73.0	0.01370
3	48.0	68.0	0.70588
4	1.0	93.0	0.01075
5	64.0	95.0	0.67368
6	64.0	98.0	0.65306
7	17.0	82.0	0.20732
8	19.0	2.0	9.50000
9	59.0	65.0	0.90769
10	20.0	95.0	0.21053
11	96.0	66.0	1.45455
12	57.0	74.0	0.77027
13	29.0	15.0	1.93333
14	70.0	5.0	14.00000
15	92.0	62.0	1.48387
16	90.0	5.0	18.00000
17	8.0	20.0	0.40000
18	75.0	94.0	0.79787
19	75.0	72.0	1.04167
20	49.0	16.0	3.06250
21	60.0	94.0	0.63830
22	65.0	38.0	1.71053
23	65.0	13.0	5.00000
24	5.0	40.0	0.12500
25	91.0	13.0	7.00000
26	82.0	69.0	1.18841
27	39.0	34.0	1.14706
28	88.0	52.0	1.69231
29	85.0	68.0	1.25000
30	76.0	82.0	0.92683
31	82.0	4.0	20.50000
32	25.0	68.0	0.36765
33	45.0	68.0	0.66176
34	30.0	85.0	0.35294
35	75.0	22.0	3.40909
36	14.0	19.0	0.73684
37	44.0	88.0	0.50000
38	90.0	55.0	1.63636
39	96.0	33.0	2.90909
40	99.0	19.0	5.21053
41	60.0	44.0	1.36364
42	71.0	57.0	1.24561
43	15.0	57.0	0.26316
44	63.0	76.0	0.82895
45	82.0	60.0	1.36667
46	17.0	26.0	0.65385
47	61.0	66.0	0.92424
48	83.0	35.0	2.37143
49	92.0	87.0	1.05747
50	41.0	30.0	1.36667

51	13.0	60.0	0.21667
52	41.0	51.0	0.80392
53	55.0	65.0	0.84615
54	24.0	10.0	2.40000
55	50.0	49.0	1.02041
56	46.0	6.0	7.66667
57	13.0	21.0	0.61905
58	16.0	3.0	5.33333
59	74.0	30.0	2.46667
60	92.0	3.0	30.66667
61	24.0	96.0	0.25000
62	36.0	89.0	0.40449
63	1.0	41.0	0.02439
64	22.0	17.0	1.29412
65	2.0	27.0	0.07407
66	51.0	63.0	0.80952
67	7.0	41.0	0.17073
68	76.0	49.0	1.55102
69	97.0	82.0	1.18293
70	45.0	24.0	1.87500
71	36.0	70.0	0.51429
72	27.0	66.0	0.40909
73	73.0	99.0	0.73737
74	20.0	72.0	0.27778
75	37.0	1.0	37.00000
76	63.0	42.0	1.50000
77	71.0	11.0	6.45455
78	84.0	20.0	4.20000
79	52.0	54.0	0.96296
80	22.0	36.0	0.61111
81	78.0	70.0	1.11429
82	17.0	23.0	0.73913
83	96.0	65.0	1.47692
84	18.0	59.0	0.30508
85	21.0	21.0	1.00000
86	40.0	57.0	0.70175
87	29.0	2.0	14.50000
88	96.0	89.0	1.07865
89	29.0	17.0	1.70588
90	21.0	37.0	0.56757
91	15.0	47.0	0.31915
92	34.0	42.0	0.80952
93	33.0	97.0	0.34021
94	6.0	48.0	0.12500
95	95.0	68.0	1.39706
96	15.0	43.0	0.34884
97	16.0	33.0	0.48485
98	6.0	21.0	0.28571
99	11.0	30.0	0.36667
**	70.0	81.0	0.86420

RIGHT-HAND-SIDE: 200.0
UPPER-BOUND ON OPTIMUM: 1165.3
PAYOFF: 1148.0 RESOURCES LEFT: 0.0

DO THESE:
8
13

```
14
16
20
23
25
31
35
39
40
54
56
60
75
77
78
87

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:    966
INITIAL SOLUTION:    17
SLIPPING:    183


RIGHT-HAND-SIDE:    500.0
UPPER-BOUND ON OPTIMUM:    1709.3
PAYOFF:    1701.0      RESOURCES LEFT:    4.0

DO THESE:
8
13
14
16
20
22
23
25
28
31
35
38
39
40
48
54
56
58
59
60
70
75
76
77
78
87
89

TIMING SUMMARY, MILLISECONDS:
```

```
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    216


RIGHT-HAND-SIDE:    1000.0
UPPER-BOUND ON OPTIMUM:    2414.7
PAYOFF:    2403.0      RESOURCES LEFT:     1.0

DO THESE:
      8
     11
     13
     14
     15
     16
     20
     22
     23
     25
     27
     28
     29
     31
     35
     38
     39
     40
     41
     48
     50
     54
     56
     58
     59
     60
     64
     68
     70
     75
     76
     77
     78
     83
     87
     89
     95


TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:     16
SLIPPING:    234


RIGHT-HAND-SIDE:    1500.0
UPPER-BOUND ON OPTIMUM:    2987.3
PAYOFF:    2984.0      RESOURCES LEFT:     2.0
```

```
DO THESE:
        8
       11
       13
       14
       15
       16
       19
       20
       22
       23
       25
       26
       27
       28
       29
       31
       35
       38
       39
       40
       41
       42
       45
       48
       50
       54
       56
       58
       59
       60
       64
       68
       69
       70
       75
       76
       77
       78
       81
       83
       87
       88
       89
       95

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    233


RIGHT-HAND-SIDE:    2000.0
UPPER-BOUND ON OPTIMUM:    3462.4
PAYOFF:    3452.3      RESOURCES LEFT:     3.0

DO THESE:
```

```
 8
 9
11
13
14
15
16
19
20
22
23
25
26
27
28
29
30
31
35
38
39
40
41
42
44
45
48
49
50
53
54
55
56
58
59
60
64
68
69
70
75
76
77
78
79
81
83
85
87
88
89
95

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:        0
SLIPPING:    217
```

```
RIGHT-HAND-SIDE:    2500.0
UPPER-BOUND ON OPTIMUM:    3864.6
PAYOFF:    3858.0      RESOURCES LEFT:    9.0

DO THESE:
        8
        9
       11
       12
       13
       14
       15
       16
       18
       19
       20
       22
       23
       25
       26
       27
       28
       29
       30
       31
       35
       38
       39
       40
       41
       42
       44
       45
       47
       48
       49
       50
       52
       53
       54
       55
       56
       58
       59
       60
       64
       66
       68
       69
       70
       75
       76
       77
       78
       79
       81
       82
       83
       85
```

```
     87
     88
     89
     92
     95
    100

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    217


RIGHT-HAND-SIDE:    3000.0
UPPER-BOUND ON OPTIMUM:    4209.3
PAYOFF:    4202.0      RESOURCES LEFT:    0.0

DO THESE:
      3
      5
      8
      9
     11
     12
     13
     14
     15
     16
     17
     18
     19
     20
     22
     23
     25
     26
     27
     28
     29
     30
     31
     33
     35
     36
     38
     39
     40
     41
     42
     44
     45
     46
     47
     48
     49
     50
     52
     53
```

```
54
55
56
57
58
59
60
64
66
68
69
70
73
75
76
77
78
79
80
81
82
83
85
86
87
88
89
92
95
100

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:    17
SLIPPING:    216


RIGHT-HAND-SIDE:    3500.0
UPPER-BOUND ON OPTIMUM:    4497.0
PAYOFF:    4488.0        RESOURCES LEFT:     5.0

DO THESE:
1
3
5
6
8
9
11
12
13
14
15
16
18
19
20
21
```

22
23
25
26
27
28
29
30
31
33
35
36
37
38
39
40
41
42
44
45
46
47
48
49
50
52
53
54
55
56
57
58
59
60
64
66
68
69
70
71
73
75
76
77
78
79
80
81
82
83
85
86
87
88
89
90
92
95
97
100

```
TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    116


RIGHT-HAND-SIDE:    4000.0
UPPER-BOUND ON OPTIMUM:    4675.2
PAYOFF:    4673.0      RESOURCES LEFT:     7.0

DO THESE:
      1
      3
      5
      6
      8
      9
     11
     12
     13
     14
     15
     16
     17
     18
     19
     20
     21
     22
     23
     25
     26
     27
     28
     29
     30
     31
     32
     33
     34
     35
     36
     37
     38
     39
     40
     41
     42
     44
     45
     46
     47
     48
     49
     50
     52
     53
```

```
      54
      55
      56
      57
      58
      59
      60
      62
      64
      66
      68
      69
      70
      71
      72
      73
      75
      76
      77
      78
      79
      80
      81
      82
      83
      85
      86
      87
      88
      89
      90
      92
      93
      95
      96
      97
      99
     100

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:     17
SLIPPING:    100


RIGHT-HAND-SIDE:    4500.0
UPPER-BOUND ON OPTIMUM:    4804.0
PAYOFF:    4804.0      RESOURCES LEFT:     0.0

DO THESE:
      1
      3
      5
      6
      8
      9
     10
     11
```

12
13
14
15
16
17
18
19
20
21
22
23
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
64
66
68
69
70
71
72
73
74
75

```
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
95
96
97
98
99
100

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:     17
SLIPPING:     33
```

THE SLIPPERY ALGORITHM

PROBLEM SET 3

VARIABLE	COEFFICIENTS OBJECTIVE	COEFFICIENTS CONSTRAINT	RATIO
1	51.0	42.0	1.21429
2	38.0	58.0	0.65517
3	19.0	21.0	0.90476
4	66.0	30.0	2.20000
5	71.0	32.0	2.21875
6	5.0	86.0	0.05814
7	13.0	5.0	2.60000
8	78.0	7.0	11.14286
9	84.0	54.0	1.55556
10	66.0	58.0	1.13793
11	93.0	74.0	1.25676
12	59.0	46.0	1.28261
13	37.0	63.0	0.58730
14	30.0	64.0	0.46875
15	6.0	46.0	0.13043
16	54.0	9.0	6.00000
17	4.0	44.0	0.09091
18	53.0	78.0	0.67949
19	62.0	83.0	0.74699
20	95.0	42.0	2.26190
21	78.0	92.0	0.84783
22	11.0	83.0	0.13253
23	81.0	80.0	1.01250
24	50.0	35.0	1.42857
25	34.0	81.0	0.41975
26	52.0	54.0	0.96296
27	81.0	150.0	0.54000
28	11.0	155.0	0.07097
29	50.0	254.0	0.19685
30	36.0	167.0	0.21557
31	61.0	117.0	0.52137
32	35.0	178.0	0.19663
33	25.0	216.0	0.11574
34	98.0	274.0	0.35766
35	93.0	163.0	0.57055
36	78.0	111.0	0.70270
37	10.0	283.0	0.03534
38	99.0	114.0	0.86842
39	10.0	103.0	0.09709
40	90.0	126.0	0.71429
41	165.0	254.0	0.64961
42	140.0	172.0	0.81395
43	151.0	140.0	1.07857
44	169.0	188.0	0.89894
45	158.0	165.0	0.95758
46	129.0	138.0	0.93478
47	128.0	142.0	0.90141
48	222.0	120.0	1.85000
49	156.0	165.0	0.94545
50	144.0	123.0	1.17073

51	183.0	101.0	1.81188
52	150.0	176.0	0.85227
53	174.0	117.0	1.48718
54	107.0	204.0	0.52451
55	8.0	87.0	0.09195
56	59.0	56.0	1.05357
57	60.0	42.0	1.42857
58	46.0	69.0	0.66667
59	64.0	11.0	5.81818
60	22.0	46.0	0.47826
61	16.0	31.0	0.51613
62	35.0	6.0	5.83333
63	85.0	100.0	0.85000
64	98.0	75.0	1.30667
65	32.0	88.0	0.36364
66	52.0	90.0	0.57778
67	16.0	15.0	1.06667
68	62.0	98.0	0.63265
69	82.0	32.0	2.56250
70	38.0	80.0	0.47500
71	73.0	17.0	4.29412
72	32.0	23.0	1.39130
73	41.0	61.0	0.67213
74	44.0	37.0	1.18919
75	96.0	73.0	1.31507
76	20.0	6.0	3.33333
77	42.0	59.0	0.71186
78	78.0	85.0	0.91765
79	5.0	100.0	0.05000
80	22.0	64.0	0.34375
81	24.0	69.0	0.34783
82	54.0	24.0	2.25000
83	35.0	83.0	0.42169
84	19.0	87.0	0.21839
85	11.0	52.0	0.21154
86	91.0	91.0	1.00000
87	60.0	83.0	0.72289
88	19.0	46.0	0.41304
89	38.0	35.0	1.08571
90	64.0	2.0	32.00000
91	14.0	49.0	0.28571
92	82.0	74.0	1.10811
93	66.0	23.0	2.86957
94	44.0	33.0	1.33333
95	100.0	97.0	1.03093
96	35.0	52.0	0.67308
97	35.0	70.0	0.50000
98	19.0	24.0	0.79167
99	63.0	18.0	3.50000
**	29.0	86.0	0.33721

RIGHT-HAND-SIDE: 500.0
UPPER-BOUND ON OPTIMUM: 1326.3
PAYOFF: 1319.0 RESOURCES LEFT: 0.0

DO THESE:
4
5

```
     7
     8
    16
    20
    48
    51
    59
    62
    67
    69
    71
    76
    82
    90
    93
    99

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:  1133
INITIAL SOLUTION:     0
SLIPPING:   200


RIGHT-HAND-SIDE:   1000.0
UPPER-BOUND ON OPTIMUM:   2021.4
PAYOFF:   2016.0      RESOURCES LEFT:    2.0

DO THESE:
     4
     5
     7
     8
     9
    12
    16
    20
    24
    48
    51
    53
    57
    59
    62
    64
    67
    69
    71
    72
    75
    76
    82
    90
    93
    94
    99

TIMING SUMMARY, MILLISECONDS:
```

```
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    216


RIGHT-HAND-SIDE:    1500.0
UPPER-BOUND ON OPTIMUM:    2597.8
PAYOFF:    2593.0      RESOURCES LEFT:     3.0

DO THESE:
     1
     4
     5
     7
     8
     9
    10
    11
    12
    16
    20
    24
    48
    50
    51
    53
    56
    57
    59
    62
    64
    67
    69
    71
    72
    74
    75
    76
    82
    89
    90
    92
    93
    94
    99


TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    266


RIGHT-HAND-SIDE:    2000.0
UPPER-BOUND ON OPTIMUM:    3107.3
PAYOFF:    3091.0      RESOURCES LEFT:     0.0

DO THESE:
```

```
     1
     3
     4
     5
     7
     8
     9
    10
    11
    12
    16
    20
    24
    26
    43
    48
    50
    51
    53
    56
    57
    59
    62
    63
    64
    67
    69
    71
    72
    74
    75
    76
    82
    86
    89
    90
    92
    93
    94
    95
    99

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:    17
SLIPPING:    216


RIGHT-HAND-SIDE:    2500.0
UPPER-BOUND ON OPTIMUM:    3578.0
PAYOFF:    3561.0      RESOURCES LEFT:     1.0

DO THESE:
     1
     3
     4
     5
     7
```

```
        8
        9
       11
       12
       16
       20
       23
       24
       26
       43
       45
       46
       48
       49
       50
       51
       53
       56
       57
       59
       62
       64
       67
       69
       71
       72
       74
       75
       76
       78
       82
       86
       89
       90
       92
       93
       94
       95
       98
       99

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:     0
SLIPPING:   217


RIGHT-HAND-SIDE:   3000.0
UPPER-BOUND ON OPTIMUM:   4023.6
PAYOFF:   4008.0     RESOURCES LEFT:   3.0

DO THESE:
       1
       3
       4
       5
       7
       8
```

```
 9
10
11
12
16
20
24
26
38
43
44
45
46
47
48
49
50
51
53
56
57
59
62
63
64
67
69
71
72
74
75
76
78
82
86
89
90
92
93
94
95
99

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:     0
SLIPPING:   217


RIGHT-HAND-SIDE:   3500.0
UPPER-BOUND ON OPTIMUM:   4443.2
PAYOFF:   4420.0      RESOURCES LEFT:     2.0

DO THESE:
 1
 3
 4
 5
```

7
8
9
10
11
12
16
19
20
21
23
24
26
38
43
44
45
46
47
48
49
50
51
52
53
56
57
59
60
62
63
64
67
69
71
72
74
75
76
78
82
86
89
90
92
93
94
95
98
99

TIMING SUMMARY MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 0
SLIPPING: 217

RIGHT-HAND-SIDE: 4000.0

UPPER-BOUND ON OPTIMUM: 4805.9
PAYOFF: 4799.0 RESOURCES LEFT: 2.0

DO THESE:

1
3
4
5
7
8
9
10
11
12
16
18
20
21
23
24
26
36
38
40
42
43
44
45
46
47
48
49
50
51
52
53
56
57
59
62
63
64
67
69
71
72
74
75
76
77
78
82
86
87
89
90
92
93
94
95

```
      98
      99

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:    17
SLIPPING:   233


RIGHT-HAND-SIDE:   5000.0
UPPER-BOUND ON OPTIMUM:   5435.9
PAYOFF:   5430.0      RESOURCES LEFT:   11.0

DO THESE:
      1
      2
      3
      4
      5
      7
      8
      9
     10
     11
     12
     13
     16
     18
     19
     20
     21
     23
     24
     26
     35
     36
     38
     40
     41
     42
     43
     44
     45
     46
     47
     48
     49
     50
     51
     52
     53
     56
     57
     58
     59
     62
     63
     64
```

66
67
68
69
71
72
73
74
75
76
77
78
82
86
87
89
90
92
93
94
95
96
98
99

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 17
SLIPPING: 150

RIGHT-HAND-SIDE: 6000.0
UPPER-BOUND ON OPTIMUM: 5922.2
PAYOFF: 5910.0 RESOURCES LEFT: 15.0

DO THESE:
1
2
3
4
5
7
8
9
10
11
12
13
16
18
19
20
21
23
24
25
26
27

31
35
36
38
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
82
83
86
87
88
89
90
92
93
94
95
96
97
98
99

TIMING SUMMARY, MILLISECONDS:

```
ORDERING VARIABLES:      0
INITIAL SOLUTION:        0
SLIPPING:    134
```

THE SLIPPERY ALGORITHM

PROBLEM SET FOUR

VARIABLE	COEFFICIENTS OBJECTIVE	CONSTRAINT	RATIO
1	48.0	68.0	0.70588
2	53.0	95.0	0.55789
3	64.0	95.0	0.67368
4	64.0	98.0	0.65306
5	59.0	65.0	0.90769
6	96.0	66.0	1.45455
7	57.0	74.0	0.77027
8	92.0	62.0	1.48387
9	75.0	94.0	0.79787
10	75.0	72.0	1.04167
11	60.0	94.0	0.63830
12	82.0	69.0	1.18841
13	39.0	34.0	1.14706
14	85.0	68.0	1.25000
15	76.0	82.0	0.92683
16	45.0	68.0	0.66176
17	14.0	19.0	0.73684
18	44.0	88.0	0.50000
19	60.0	44.0	1.36364
20	71.0	57.0	1.24561
21	63.0	76.0	0.82895
22	82.0	60.0	1.36667
23	17.0	26.0	0.65385
24	61.0	66.0	0.92424
25	92.0	87.0	1.05747
26	41.0	30.0	1.36667
27	41.0	51.0	0.80392
28	55.0	65.0	0.84615
29	50.0	49.0	1.02041
30	13.0	21.0	0.61905
31	22.0	17.0	1.29412
32	51.0	63.0	0.80952
33	97.0	82.0	1.18293
34	36.0	70.0	0.51429
35	73.0	99.0	0.73737
36	63.0	42.0	1.50000
37	52.0	54.0	0.96296
38	22.0	36.0	0.61111
39	17.0	23.0	0.73913
40	21.0	21.0	1.00000
41	40.0	57.0	0.70175
42	96.0	89.0	1.07865
43	21.0	37.0	0.56757
44	34.0	42.0	0.80952
45	95.0	68.0	1.39706
46	70.0	81.0	0.86420
47	51.0	42.0	1.21429
48	38.0	58.0	0.65517
49	19.0	21.0	0.90476
50	66.0	58.0	1.13793

51	93.0	74.0	1.25676
52	59.0	46.0	1.28261
53	37.0	63.0	0.58730
54	53.0	78.0	0.67949
55	62.0	83.0	0.74699
56	78.0	92.0	0.84783
57	81.0	80.0	1.01250
58	50.0	35.0	1.42857
59	52.0	54.0	0.96296
60	81.0	150.0	0.54000
61	61.0	117.0	0.52137
62	93.0	163.0	0.57055
63	78.0	111.0	0.70270
64	99.0	114.0	0.86842
65	97.0	126.0	0.71429
66	165.0	254.0	0.64961
67	140.0	172.0	0.81395
68	151.0	140.0	1.07857
69	169.0	188.0	0.89894
70	158.0	165.0	0.95758
71	129.0	138.0	0.93478
72	128.0	142.0	0.90141
73	156.0	165.0	0.94545
74	144.0	123.0	1.17073
75	150.0	176.0	0.85227
76	174.0	117.0	1.48718
77	107.0	204.0	0.52451
78	59.0	56.0	1.05357
79	60.0	42.0	1.42857
80	46.0	69.0	0.66667
81	16.0	31.0	0.51613
82	85.0	100.0	0.85000
83	98.0	75.0	1.30667
84	52.0	90.0	0.57778
85	16.0	15.0	1.06667
86	62.0	98.0	0.63265
87	32.0	23.0	1.39130
88	41.0	61.0	0.67213
89	44.0	37.0	1.18919
90	96.0	73.0	1.31507
91	42.0	59.0	0.71186
92	78.0	85.0	0.91765
93	91.0	91.0	1.00000
94	60.0	83.0	0.72289
95	38.0	35.0	1.08571
96	82.0	74.0	1.10811
97	44.0	33.0	1.33333
98	100.0	97.0	1.03093
99	35.0	52.0	0.67308
**	19.0	24.0	0.79167

RIGHT-HAND-SIDE: 500.0
UPPER-BOUND ON OPTIMUM: 723.5
PAYOFF: 719.0 RESOURCES LEFT: 0.0

DO THESE:
6
8

```
     26
     36
     45
     56
     76
     79
     85
     87

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:    983
INITIAL SOLUTION:      0
SLIPPING:    184


RIGHT-HAND-SIDE:    1000.0
UPPER-BOUND ON OPTIMUM:    1373.1
PAYOFF:    1359.0      RESOURCES LEFT:    3.0

DO THESE:
      6
      8
     14
     19
     22
     26
     31
     36
     45
     47
     51
     52
     76
     79
     83
     85
     87
     90
     97


TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:    17
SLIPPING:    183


RIGHT-HAND-SIDE:    1500.0
UPPER-BOUND ON OPTIMUM:    1961.5
PAYOFF:    1957.7      RESOURCES LEFT:    2.0

DO THESE:
      6
      8
     12
     13
     14
```

```
19
20
22
26
31
33
36
40
45
47
50
51
52
58
74
76
79
83
87
89
90
97

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    216


RIGHT-HAND-SIDE:    2000.0
UPPER-BOUND ON OPTIMUM:    2496.1
PAYOFF:    2494.0      RESOURCES LEFT:     0.0

DO THESE:
 6
 8
12
13
14
19
20
22
25
26
31
33
36
40
42
45
47
49
50
51
52
58
68
74
```

```
76
78
79
83
87
89
90
95
96
97


TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:     0
SLIPPING:    216


RIGHT-HAND-SIDE:    2500.0
UPPER-BOUND ON OPTIMUM:    3200.6
PAYOFF:    2995.0      RESOURCES LEFT:     3.0

DO THESE:
  6
  8
 10
 12
 13
 14
 19
 20
 22
 25
 26
 29
 31
 33
 36
 37
 40
 42
 45
 47
 49
 50
 51
 52
 57
 58
 59
 68
 74
 76
 78
 79
 83
 87
 89
 90
```

```
93
95
96
97
98

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    233


RIGHT-HAND-SIDE:    3000.0
UPPER-BOUND ON OPTIMUM:    3473.0
PAYOFF:    3468.0      RESOURCES LEFT:     1.0

DO THESE:
    6
    8
   10
   12
   13
   14
   17
   19
   20
   22
   25
   26
   29
   31
   33
   36
   37
   40
   42
   45
   47
   49
   50
   51
   52
   57
   58
   59
   68
   70
   71
   73
   74
   76
   78
   79
   83
   85
   87
   89
   90
```

```
     93
     95
     96
     97
     98

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    216


RIGHT-HAND-SIDE:    3500.0
UPPER-BOUND ON OPTIMUM:    3927.9
PAYOFF:    3920.0      RESOURCES LEFT:     0.0

DO THESE:
      5
      6
      8
     10
     12
     13
     14
     15
     19
     20
     22
     25
     26
     28
     29
     31
     33
     36
     37
     40
     42
     45
     46
     47
     49
     50
     51
     52
     57
     58
     59
     68
     70
     71
     72
     73
     74
     76
     78
     79
     83
```

```
      85
      87
      89
      90
      92
      93
      95
      96
      97
      98

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    200


RIGHT-HAND-SIDE:    4000.0
UPPER-BOUND ON OPTIMUM:    4361.8
PAYOFF:    4358.0      RESOURCES LEFT:    2.0

DO THESE:
      5
      6
      8
     10
     12
     13
     14
     15
     17
     19
     20
     22
     24
     25
     26
     29
     31
     33
     36
     37
     40
     42
     45
     46
     47
     49
     50
     51
     52
     57
     58
     59
     64
     68
     69
     70
```

```
71
72
73
74
75
76
78
79
83
85
87
89
90
92
93
95
96
97
98

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:     0
SLIPPING:    200


RIGHT-HAND-SIDE:    5000.0
UPPER-BOUND ON OPTIMUM:    5166.7
PAYOFF:    5163.0      RESOURCES LEFT:     1.0

DO THESE:
5
6
7
8
9
10
12
13
14
15
17
19
20
21
22
23
24
25
26
27
28
29
31
32
33
35
36
```

```
 37
 39
 40
 42
 44
 45
 46
 47
 49
 50
 51
 52
 56
 57
 58
 59
 64
 67
 68
 69
 70
 71
 72
 73
 74
 75
 76
 78
 79
 82
 83
 85
 87
 89
 90
 92
 93
 95
 96
 97
 98
100

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    167


RIGHT-HAND-SIDE:    6000.0
UPPER-BOUND ON OPTIMUM:    5961.1
PAYOFF:    5854.0      RESOURCES LEFT:    1.0

DO THESE:
  1
  3
  5
  6
```

7
8
9
10
12
13
14
15
16
17
19
20
21
22
24
25
26
27
28
29
31
32
33
35
36
37
39
40
41
42
45
46
47
48
49
50
51
52
54
55
56
57
58
59
63
64
65
67
68
69
70
71
72
73
74
75
76
78
79
80

```
 82
 83
 85
 87
 88
 89
 90
 91
 92
 93
 94
 95
 96
 97
 98
 99
100

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:     17
SLIPPING:    100
```

THE SLIPPERY ALGORITHM

PROBLEM SET FIVE

VARIABLE	COEFFICIENTS OBJECTIVE	CONSTRAINT	RATIO
1	48.0	68.0	0.70588
2	1.0	73.0	0.01370
3	17.0	82.0	0.20732
4	59.0	65.0	0.90769
5	96.0	66.0	1.45455
6	57.0	74.0	0.77027
7	92.0	62.0	1.48387
8	75.0	72.0	1.04167
9	65.0	38.0	1.71053
10	5.0	40.0	0.12500
11	82.0	69.0	1.18841
12	88.0	52.0	1.69231
13	85.0	68.0	1.25000
14	76.0	82.0	0.92683
15	25.0	68.0	0.36765
16	45.0	68.0	0.66176
17	30.0	85.0	0.35294
18	90.0	55.0	1.63636
19	96.0	33.0	2.90909
20	60.0	44.0	1.36364
21	71.0	57.0	1.24561
22	15.0	57.0	0.26316
23	63.0	76.0	0.82895
24	82.0	60.0	1.36667
25	61.0	66.0	0.92424
26	83.0	35.0	2.37143
27	41.0	30.0	1.36667
28	13.0	60.0	0.21667
29	41.0	51.0	0.80392
30	55.0	65.0	0.84615
31	50.0	49.0	1.02041
32	74.0	30.0	2.46667
33	1.0	41.0	0.02439
34	51.0	63.0	0.80952
35	7.0	41.0	0.17073
36	76.0	49.0	1.55102
37	97.0	82.0	1.18293
38	36.0	70.0	0.51429
39	27.0	66.0	0.40909
40	20.0	72.0	0.27778
41	63.0	42.0	1.50000
42	52.0	54.0	0.96296
43	22.0	36.0	0.61111
44	78.0	70.0	1.11429
45	96.0	65.0	1.47692
46	18.0	59.0	0.30508
47	40.0	57.0	0.70175
48	21.0	37.0	0.56757
49	15.0	47.0	0.31915
50	34.0	42.0	0.80952

51	6.0	48.0	0.12500
52	95.0	68.0	1.39706
53	15.0	43.0	0.34884
54	16.0	33.0	0.48485
55	11.0	30.0	0.36667
56	70.0	81.0	0.86420
57	51.0	42.0	1.21429
58	38.0	58.0	0.65517
59	66.0	30.0	2.20000
60	71.0	32.0	2.21875
61	84.0	54.0	1.55556
62	66.0	58.0	1.13793
63	93.0	74.0	1.25676
64	59.0	46.0	1.28261
65	37.0	63.0	0.58730
66	30.0	64.0	0.46875
67	6.0	46.0	0.13043
68	4.0	44.0	0.09091
69	53.0	78.0	0.67949
70	62.0	83.0	0.74699
71	95.0	42.0	2.26190
72	11.0	83.0	0.13253
73	81.0	80.0	1.01250
74	50.0	35.0	1.42857
75	34.0	81.0	0.41975
76	52.0	54.0	0.96296
77	59.0	56.0	1.05357
78	60.0	42.0	1.42857
79	46.0	69.0	0.66667
80	22.0	46.0	0.47826
81	16.0	31.0	0.51613
82	98.0	75.0	1.30667
83	82.0	32.0	2.56250
84	38.0	80.0	0.47500
85	41.0	61.0	0.67213
86	44.0	37.0	1.18919
87	96.0	73.0	1.31507
88	78.0	85.0	0.91765
89	22.0	64.0	0.34375
90	24.0	69.0	0.34783
91	35.0	83.0	0.42169
92	11.0	52.0	0.21154
93	60.0	83.0	0.72289
94	19.0	46.0	0.41304
95	38.0	35.0	1.08571
96	14.0	49.0	0.28571
97	82.0	74.0	1.10811
98	44.0	33.0	1.33333
99	35.0	52.0	0.67308
**	35.0	70.0	0.50000

RIGHT-HAND-SIDE: 200.0
UPPER-BOUND ON OPTIMUM: 492.1
PAYOFF: 438.0 RESOURCES LEFT: 5.0

DO THESE:
18
19

```
     26
     32
     71

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:  1183
INITIAL SOLUTION:    17
SLIPPING:   183


RIGHT-HAND-SIDE:     500.0
UPPER-BOUND ON OPTIMUM:     997.0
PAYOFF:     978.0      RESOURCES LEFT:     5.0

DO THESE:
     9
    18
    19
    26
    32
    36
    45
    59
    60
    61
    71
    83

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:     0
SLIPPING:   200


RIGHT-HAND-SIDE:   1000.0
UPPER-BOUND ON OPTIMUM:   1710.3
PAYOFF:   1708.3      RESOURCES LEFT:    1.0

DO THESE:
     5
     7
     9
    12
    18
    19
    20
    24
    26
    32
    36
    41
    45
    52
    59
    62
    61
```

```
      71
      74
      78
      83
      98

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:    17
SLIPPING:   217


RIGHT-HAND-SIDE:   1500.0
UPPER-BOUND ON OPTIMUM:   2344.8
PAYOFF:   2340.0      RESOURCES LEFT:    1.0

DO THESE:
       5
       7
       9
      12
      13
      18
      19
      20
      21
      24
      26
      27
      32
      36
      41
      45
      52
      57
      59
      60
      61
      63
      64
      71
      74
      78
      82
      83
      87
      95
      98

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:   217


RIGHT-HAND-SIDE:   2000.0
```

```
UPPER-BOUND ON OPTIMUM:    2905.3
PAYOFF:    2900.0      RESOURCES LEFT:     1.0

DO THESE:
      5
      7
      9
     11
     12
     13
     18
     19
     20
     21
     24
     26
     27
     32
     36
     37
     41
     44
     45
     52
     57
     59
     60
     61
     62
     63
     64
     71
     74
     76
     77
     78
     82
     83
     86
     87
     95
     97
     98

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    233

RIGHT-HAND-SIDE:    2500.0
UPPER-BOUND ON OPTIMUM:    3384.8
PAYOFF:    3380.0      RESOURCES LEFT:     2.0

DO THESE:
      4
      5
      7
```

```
        8
        9
       11
       12
       13
       14
       18
       19
       20
       21
       24
       25
       26
       27
       31
       32
       36
       37
       41
       42
       44
       45
       52
       57
       59
       60
       61
       62
       63
       64
       71
       73
       74
       77
       78
       82
       83
       86
       87
       88
       95
       97
       98

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:     16
SLIPPING:    234

RIGHT-HAND-SIDE:    3000.0
UPPER-BOUND ON OPTIMUM:    3799.9
PAYOFF:    3794.0      RESOURCES LEFT:     2.0

DO THESE:
       1
       4
       5
```

7
8
9
11
12
13
14
18
19
20
21
23
24
25
26
27
29
30
31
32
34
36
37
41
42
44
45
50
52
56
57
59
60
61
62
63
64
71
73
74
76
77
78
82
83
86
87
88
95
97
98

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 17
SLIPPING: 216

```
RIGHT-HAND-SIDE:    3500.0
UPPER-BOUND ON OPTIMUM:    4151.3
PAYOFF:    4148.0      RESOURCES LEFT:     2.0

DO THESE:
      1
      4
      5
      6
      7
      8
      9
     11
     12
     13
     14
     18
     19
     20
     21
     23
     24
     25
     26
     27
     29
     30
     31
     32
     34
     36
     37
     41
     42
     44
     45
     50
     52
     56
     57
     59
     60
     61
     62
     63
     64
     69
     70
     71
     73
     74
     76
     77
     78
     79
     82
     83
     85
     86
     87
```

```
      88
      93
      95
      97
      98
      99

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    200


RIGHT-HAND-SIDE:    4000.0
UPPER-BOUND ON OPTIMUM:    4443.8
PAYOFF:    4439.0      RESOURCES LEFT:    3.0

DO THESE:
      1
      4
      5
      6
      7
      8
      9
     11
     12
     13
     14
     16
     18
     19
     20
     21
     23
     24
     25
     26
     27
     29
     30
     31
     32
     34
     36
     37
     38
     41
     42
     43
     44
     45
     47
     50
     52
     56
     57
     58
```

59
60
61
62
63
64
65
69
70
71
73
74
76
77
78
79
80
81
82
83
85
86
87
88
93
95
97
98
99
100

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 0
SLIPPING: 150

RIGHT-HAND-SIDE: 4500.0
UPPER-BOUND ON OPTIMUM: 4663.8
PAYOFF: 4659.0 RESOURCES LEFT: 13.0

DO THESE:
1
4
5
6
7
8
9
11
12
13
14
16
18
19
20
21

23
24
25
26
27
29
30
31
32
34
36
37
38
39
41
42
43
44
45
47
48
50
52
54
56
57
58
59
60
61
62
63
64
65
66
69
70
71
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
91
93
94
95
97
96

99
100

```
TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    167
```

THE SLIPPERY ALGORITHM

PROBLEM SET SIX

VARIABLE	COEFFICIENTS OBJECTIVE	CONSTRAINT	RATIO
1	54.0	541.0	0.09982
2	68.0	786.0	0.08651
3	789.0	3541.0	0.22282
4	65.0	75.0	0.86667
5	24.0	1.0	24.00000
6	35.0	652.0	0.05368
7	61.0	843.0	0.07236
8	42.0	21.0	2.00000
9	786.0	10.0	78.60000
10	43.0	40.0	1.07500
11	512.0	46.0	11.13043
12	424.0	51.0	8.31373
13	321.0	752.0	0.42686
14	11.0	810.0	0.01358
15	4.0	510.0	0.00784
16	47.0	21.0	2.23810
17	35.0	42.0	0.83333
18	56.0	121.0	0.46281
19	12.0	5.0	2.40000
20	754.0	4.0	188.50000
21	55.0	72.0	0.76389
22	457.0	631.0	0.72425
23	1.0	720.0	0.00139
24	65.0	435.0	0.14943
25	74.0	2.0	37.00000
26	22.0	820.0	0.02683
27	4.0	64.0	0.06250
28	45.0	73.0	0.61644
29	12.0	770.0	0.01558
30	5.0	43.0	0.11628
31	57.0	85.0	0.67059
32	57.0	912.0	0.06250
33	23.0	4.0	5.75000
34	54.0	35.0	1.54286
35	2.0	14.0	0.14286
36	42.0	42.0	1.00000
37	77.0	22.0	3.50000
38	78.0	54.0	1.44444
39	682.0	32.0	21.31250
40	142.0	35.0	4.05714
41	150.0	175.0	0.85714
42	1200.0	1200.0	1.00000
43	400.0	300.0	1.33333
44	500.0	350.0	1.42857
45	98.0	12.0	8.16667
46	48.0	68.0	0.70588
47	1.0	73.0	0.01370
48	17.0	82.0	0.20732
49	59.0	65.0	0.90769
50	96.0	66.0	1.45455

51	57.0	74.0	0.77027
52	92.0	62.0	1.48387
53	75.0	72.0	1.04167
54	65.0	38.0	1.71053
55	5.0	40.0	0.12500
56	82.0	69.0	1.18841
57	88.0	52.0	1.69231
58	85.0	68.0	1.25000
59	76.0	82.0	0.92683
60	25.0	68.0	0.36765
61	45.0	68.0	0.66176
62	30.0	85.0	0.35294
63	90.0	55.0	1.63636
64	96.0	33.0	2.90909
65	60.0	44.0	1.36364
66	71.0	57.0	1.24561
67	15.0	57.0	0.26316
68	63.0	76.0	0.82895
69	82.0	60.0	1.36667
70	61.0	66.0	0.92424
71	83.0	35.0	2.37143
72	41.0	30.0	1.36667
73	13.0	60.0	0.21667
74	41.0	51.0	0.80392
75	55.0	65.0	0.84615
76	50.0	49.0	1.02041
77	74.0	30.0	2.46667
78	1.0	41.0	0.02439
79	51.0	63.0	0.80952
80	7.0	41.0	0.17073
81	76.0	49.0	1.55102
82	97.0	82.0	1.18293
83	36.0	70.0	0.51429
84	27.0	66.0	0.40909
85	20.0	72.0	0.27778
86	63.0	42.0	1.50000
87	52.0	54.0	0.96296
88	22.0	36.0	0.61111
89	78.0	70.0	1.11429
90	96.0	65.0	1.47692
91	18.0	59.0	0.30508
92	40.0	57.0	0.70175
93	21.0	37.0	0.56757
94	15.0	47.0	0.31915
95	34.0	42.0	0.80952
96	6.0	48.0	0.12500
97	95.0	68.0	1.39706
98	15.0	43.0	0.34884
99	16.0	33.0	0.48485
100	11.0	30.0	0.36667
101	70.0	81.0	0.86420
102	51.0	42.0	1.21429
103	38.0	58.0	0.65517
104	66.0	30.0	2.20000
105	71.0	32.0	2.21875
106	84.0	54.0	1.55556
107	66.0	58.0	1.13793
108	93.0	74.0	1.25676
109	59.0	46.0	1.28261
110	37.0	63.0	0.58730

111	30.0	64.0	0.46875
112	6.0	46.0	0.13043
113	4.0	44.0	0.09091
114	53.0	78.0	0.67949
115	62.0	83.0	0.74699
116	95.0	42.0	2.26190
117	11.0	83.0	0.13253
118	81.0	80.0	1.01250
119	50.0	35.0	1.42857
120	34.0	81.0	0.41975
121	52.0	54.0	0.96296
122	59.0	56.0	1.05357
123	60.0	42.0	1.42857
124	46.0	69.0	0.66667
125	22.0	46.0	0.47826
126	16.0	31.0	0.51613
127	98.0	75.0	1.30667
128	82.0	32.0	2.56250
129	38.0	80.0	0.47500
130	41.0	61.0	0.67213
131	44.0	37.0	1.18919
132	96.0	73.0	1.31507
133	78.0	85.0	0.91765
134	22.0	64.0	0.34375
135	24.0	69.0	0.34783
136	35.0	83.0	0.42169
137	11.0	52.0	0.21154
138	60.0	83.0	0.72289
139	19.0	46.0	0.41304
140	38.0	35.0	1.08571
141	14.0	49.0	0.28571
142	82.0	74.0	1.10811
143	44.0	33.0	1.33333
144	35.0	52.0	0.67308
145	35.0	70.0	0.50000
146	11.0	155.0	0.07097
147	50.0	254.0	0.19685
148	36.0	167.0	0.21557
149	35.0	178.0	0.19663
150	183.0	101.0	1.81188

RIGHT-HAND-SIDE: 500.0
UPPER-BOUND ON OPTIMUM: 4264.0
PAYOFF: 4264.0 RESOURCES LEFT: 0.0

DO THESE:
5
8
9
11
12
16
19
20
25
33
37
39

```
  40
  45
  64
  71
  77
 104
 105
 116
 128

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:  1667
INITIAL SOLUTION:     0
SLIPPING:   400


RIGHT-HAND-SIDE:   1000.0
UPPER-BOUND ON OPTIMUM:   5076.7
PAYOFF:   5065.0     RESOURCES LEFT:   2.0

DO THESE:
   5
   8
   9
  11
  12
  16
  19
  20
  25
  33
  34
  37
  39
  40
  45
  52
  54
  57
  64
  71
  77
  81
  86
  90
 104
 105
 106
 116
 128
 150

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:     0
SLIPPING:   466
```

```
RIGHT-HAND-SIDE:   1500.0
UPPER-BOUND ON OPTIMUM:   5796.1
PAYOFF:   5763.0      RESOURCES LEFT:     1.0

DO THESE:
      5
      8
      9
     11
     12
     16
     19
     20
     25
     33
     34
     35
     37
     38
     39
     40
     45
     50
     52
     54
     57
     63
     64
     65
     69
     71
     72
     77
     81
     86
     90
     97
    104
    105
    106
    116
    119
    123
    128
    143
    150


TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:       0
SLIPPING:   466


RIGHT-HAND-SIDE:   2000.0
UPPER-BOUND ON OPTIMUM:   6481.0
PAYOFF:   6457.0      RESOURCES LEFT:     3.0
```

```
DO THESE:
     5
     8
     9
    11
    12
    16
    19
    20
    25
    33
    34
    35
    37
    38
    39
    40
    44
    45
    50
    52
    54
    57
    63
    64
    65
    69
    71
    72
    77
    81
    86
    90
    97
   104
   105
   106
   116
   119
   123
   127
   128
   132
   143
   150

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    466


RIGHT-HAND-SIDE:    2500.0
UPPER-BOUND ON OPTIMUM:    7128.1
PAYOFF:    7102.3      RESOURCES LEFT:     0.0

DO THESE:
```

```
  5
  8
  9
 11
 12
 16
 19
 20
 25
 33
 34
 35
 37
 38
 39
 40
 43
 44
 45
 50
 52
 54
 57
 58
 63
 64
 65
 66
 69
 71
 72
 77
 81
 90
 97
104
105
106
108
109
116
119
123
127
128
132
143
150

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:     17
SLIPPING:    550

RIGHT-HAND-SIDE:    3000.0
UPPER-BOUND ON OPTIMUM:    7706.4
PAYOFF:    7706.4      RESOURCES LEFT:     0.0
```

DO THESE:

5
8
9
10
11
12
16
19
20
25
33
34
37
38
39
40
43
44
45
50
52
54
56
57
58
63
64
65
66
69
71
72
77
81
82
86
89
90
97
102
104
105
106
107
108
109
116
119
123
127
128
131
132
142
143
150

```
TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:    17
SLIPPING:    466


RIGHT-HAND-SIDE:    3500.0
UPPER-BOUND ON OPTIMUM:    8217.0
PAYOFF:    8192.0      RESOURCES LEFT:     2.0

DO THESE:
      5
      8
      9
     10
     11
     12
     16
     17
     19
     20
     25
     33
     34
     35
     36
     37
     38
     39
     40
     43
     44
     45
     50
     52
     53
     54
     56
     57
     58
     63
     64
     65
     66
     69
     71
     72
     76
     77
     81
     82
     86
     87
     89
     90
     97
    102
    104
    105
```

```
106
107
108
109
116
118
119
121
122
123
127
128
131
132
140
142
143
150

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:   500


RIGHT-HAND-SIDE:   4000.0
UPPER-BOUND ON OPTIMUM:   8717.0
PAYOFF:   8641.0      RESOURCES LEFT:     3.0

DO THESE:
  4
  5
  8
  9
 10
 11
 12
 16
 19
 20
 25
 33
 34
 36
 37
 38
 39
 40
 43
 44
 45
 49
 50
 52
 53
 54
 56
 57
```

```
    58
    59
    63
    64
    65
    66
    69
    70
    71
    72
    75
    76
    77
    81
    82
    86
    87
    88
    89
    90
    97
   101
   102
   104
   105
   106
   107
   108
   109
   116
   118
   119
   121
   122
   123
   127
   128
   131
   132
   133
   140
   142
   143
   150

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:     17
SLIPPING:    466


RIGHT-HAND-SIDE:    5000.0
UPPER-BOUND ON OPTIMUM:    9681.0
PAYOFF:    9666.0      RESOURCES LEFT:    4.0

DO THESE:
    5
    8
```

9
10
11
12
16
17
19
20
25
33
34
35
36
37
38
39
40
42
43
44
45
49
50
52
53
54
56
57
58
59
63
64
65
66
69
70
71
72
76
77
81
82
86
87
89
90
97
102
104
105
106
107
108
109
116
118
119
121
122
123

```
127
128
131
132
133
140
142
143
150

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:     0
SLIPPING:    467


RIGHT-HAND-SIDE:    6000.0
UPPER-BOUND ON OPTIMUM:  10483.6
PAYOFF:  10480.0      RESOURCES LEFT:     0.0

DO THESE:
     4
     5
     8
     9
    10
    11
    12
    16
    17
    19
    20
    21
    25
    33
    34
    36
    37
    38
    39
    40
    41
    42
    43
    44
    45
    49
    50
    51
    52
    53
    54
    56
    57
    58
    59
    63
    64
```

```
 65
 66
 68
 69
 70
 71
 72
 74
 75
 76
 77
 79
 81
 82
 86
 87
 89
 90
 95
 97
101
102
104
105
106
107
108
109
114
115
116
118
119
121
122
123
127
128
131
132
133
138
140
142
143
150

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    467


RIGHT-HAND-SIDE:    7000.0
UPPER-BOUND ON OPTIMUM:  11188.7
PAYOFF:  11185.0      RESOURCES LEFT:     2.0

DO THESE:
```

4
5
8
9
10
11
12
16
17
19
20
21
22
25
31
33
34
36
37
38
39
40
41
42
43
44
45
46
49
50
51
52
53
54
56
57
58
59
63
64
65
66
68
69
70
71
72
74
75
76
77
79
81
82
86
87
88
89
90
92

```
 95
 97
101
102
104
105
106
107
108
109
114
115
116
118
119
121
122
123
124
127
128
131
132
133
138
140
142
143
144
150

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:    16
SLIPPING:   434


RIGHT-HAND-SIDE:   8000.0
UPPER-BOUND ON OPTIMUM:  11715.2
PAYOFF:  11709.0     RESOURCES LEFT:    0.0

DO THESE:
  4
  5
  8
  9
 10
 11
 12
 16
 17
 18
 19
 20
 21
 22
 25
 28
```

31
33
34
35
36
37
38
39
40
41
42
43
44
45
46
49
50
51
52
53
54
56
57
58
59
61
63
64
65
66
68
69
70
71
72
74
75
76
77
79
81
82
83
86
87
88
89
90
92
93
95
97
99
100
101
102
103
104
105
106

```
107
108
109
110
111
114
115
116
118
119
121
122
123
124
125
126
127
128
129
130
131
132
133
136
138
140
142
143
144
145
150
```

```
TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:     0
SLIPPING:   334
```

THE SLIPPERY ALGORITHM

PROBLEM SET SEVEN

VARIABLE	COEFFICIENTS OBJECTIVE	COEFFICIENTS CONSTRAINT	RATIO
1	312.0	418.0	0.74641
2	195.0	112.0	1.74107
3	250.0	275.0	0.90909
4	400.0	250.0	1.60000
5	600.0	900.0	0.66667
6	1000.0	830.0	1.20482
7	100.0	4.0	25.00000
8	555.0	600.0	0.92500
9	54.0	541.0	0.09982
10	68.0	786.0	0.08651
11	789.0	3541.0	0.22282
12	65.0	75.0	0.86667
13	24.0	1.0	24.00000
14	35.0	652.0	0.05368
15	61.0	843.0	0.07236
16	42.0	21.0	2.00000
17	786.0	10.0	78.60000
18	43.0	40.0	1.07500
19	512.0	46.0	11.13043
20	424.0	51.0	8.31373
21	321.0	752.0	0.42686
22	11.0	810.0	0.01358
23	4.0	510.0	0.00784
24	47.0	21.0	2.23810
25	35.0	42.0	0.83333
26	56.0	121.0	0.46281
27	12.0	5.0	2.40000
28	754.0	4.0	188.50000
29	55.0	72.0	0.76389
30	457.0	631.0	0.72425
31	1.0	720.0	0.00139
32	65.0	435.0	0.14943
33	74.0	2.0	37.00000
34	22.0	820.0	0.02683
35	4.0	64.0	0.06250
36	45.0	73.0	0.61644
37	12.0	770.0	0.01558
38	5.0	43.0	0.11628
39	57.0	85.0	0.67059
40	57.0	912.0	0.06250
41	23.0	4.0	5.75000
42	54.0	35.0	1.54286
43	2.0	14.0	0.14286
44	42.0	42.0	1.00000
45	77.0	22.0	3.50000
46	78.0	54.0	1.44444
47	682.0	32.0	21.31250
48	142.0	35.0	4.05714
49	150.0	175.0	0.85714
50	125.0	125.0	1.00000

51	400.0	300.0	1.33333
52	500.0	350.0	1.42857
53	98.0	12.0	8.16667
54	48.0	68.0	0.70588
55	1.0	73.0	0.01370
56	17.0	82.0	0.20732
57	59.0	65.0	0.90769
58	96.0	66.0	1.45455
59	57.0	74.0	0.77027
60	92.0	62.0	1.48387
61	75.0	72.0	1.04167
62	65.0	38.0	1.71053
63	5.0	40.0	0.12500
64	82.0	69.0	1.18841
65	88.0	52.0	1.69231
66	85.0	68.0	1.25000
67	76.0	82.0	0.92683
68	25.0	68.0	0.36765
69	45.0	68.0	0.66176
70	30.0	85.0	0.35294
71	90.0	55.0	1.63636
72	96.0	33.0	2.90909
73	60.0	44.0	1.36364
74	71.0	57.0	1.24561
75	15.0	57.0	0.26316

RIGHT-HAND-SIDE: 1500.0
UPPER-BOUND ON OPTIMUM: 5716.3
PAYOFF: 5696.3 RESOURCES LEFT: 11.0

DO THESE:

2
4
7
13
16
17
19
20
24
27
28
33
41
42
45
46
47
48
52
53
58
60
62
65
66
71
72

```
TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:    700
INITIAL SOLUTION:      0
SLIPPING:    116


RIGHT-HAND-SIDE:    2000.0
UPPER-BOUND ON OPTIMUM:    6352.5
PAYOFF:    6334.0      RESOURCES LEFT:     3.0

DO THESE:
      2
      4
      7
     13
     16
     17
     18
     19
     20
     24
     27
     28
     33
     41
     42
     44
     45
     46
     47
     48
     51
     52
     53
     58
     60
     62
     64
     65
     66
     71
     72
     73
     74


TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:    3
INITIAL SOLUTION:      0
SLIPPING:    116


RIGHT-HAND-SIDE:    2500.0
UPPER-BOUND ON OPTIMUM:    6955.0
PAYOFF:    6794.0      RESOURCES LEFT:     9.0
```

```
DO THESE:
     2
     3
     4
     7
    13
    16
    17
    18
    19
    22
    24
    27
    28
    33
    41
    42
    44
    45
    46
    47
    48
    51
    52
    53
    57
    58
    60
    61
    62
    64
    65
    66
    67
    71
    72
    73
    74

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:    17
SLIPPING:   100


RIGHT-HAND-SIDE;   3000.0
UPPER-BOUND OF OPTIMUM:   7512.0
PAYOFF:   7487.0    RESOURCES LEFT:    5.0

DO THESE:
     2
     4
     6
     7
    13
    16
    17
```

```
18
19
20
24
27
28
33
41
42
43
44
45
46
47
48
51
52
53
58
60
61
62
64
65
66
67
71
72
73
74

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:     0
SLIPPING:    100


RIGHT-HAND-SIDE:    3100.0
UPPER-BOUND OF OPTIMUM:    7610.0
PAYOFF:    7585.0      RESOURCES LEFT:     2.0

DO THESE:
2
4
6
7
12
13
16
17
18
19
20
24
25
27
28
33
```

```
41
42
44
45
46
47
48
51
52
53
58
60
61
62
64
65
66
67
71
72
73
74

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:     0
SLIPPING:   117


RIGHT-HAND-SIDE:   8200.0
UPPER-BOUND ON OPTIMUM:   7710.0
PAYOFF:   7646.      RESOURCES LEFT:   23.0

DO THESE:
2
4
6
7
12
13
16
17
18
19
20
24
25
27
28
33
41
42
43
44
45
46
47
48
```

```
51
52
53
57
58
60
61
62
64
65
66
67
71
72
73
74

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:     0
SLIPPING:    100


RIGHT-HAND-SIDE:    3300.0
UPPER-BOUND ON OPTIMUM:    7810.0
PAYOFF:    7770.0      RESOURCES LEFT:    2.0

DO THESE:
2
3
4
6
7
13
16
17
18
19
20
24
25
27
28
33
41
42
44
45
46
47
48
51
52
53
58
60
61
62
```

```
64
65
66
67
71
72
73
74

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    120

RIGHT-HAND-SIDE:    3400.0
UPPER-BOUND ON OPTIMUM:    7910.0
PAYOFF:    7859.0      RESOURCES LEFT:     4.0

DO THESE:
2
3
4
6
7
12
13
16
17
18
19
20
24
27
28
33
41
42
44
45
46
47
48
51
52
53
57
58
60
61
62
64
65
66
67
71
72
73
```

```
TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:        0
SLIPPING:    116


RIGHT-HAND-SIDE:    3500.0
UPPER-BOUND OF OPTIMUM:    8210.0
PAYOFF:    7929.0      RESOURCES LEFT:    5.0

DO THESE:
     2
     3
     4
     6
     7
    12
    13
    16
    17
    18
    19
    20
    24
    25
    27
    28
    33
    41
    42
    44
    45
    46
    47
    48
    51
    52
    53
    57
    58
    60
    61
    62
    64
    65
    66
    67
    71
    72
    73
    74
    75


TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
```

```
INITIAL SOLUTION:      0
SLIPPING:    116

RIGHT-HAND-SIDE:    4000.2
UPPER-BOUND ON OPTIMUM:    8510.0
PAYOFF:    8414.2      RESOURCES LEFT:    4.0

DO THESE:
      2
      3
      4
      6
      7
      8
     12
     13
     16
     17
     18
     19
     20
     24
     27
     28
     33
     41
     42
     44
     45
     46
     47
     48
     51
     52
     53
     57
     58
     60
     61
     62
     64
     65
     66
     67
     71
     72
     73
     74

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:      0
SLIPPING:    140

RIGHT-HAND-SIDE:    4500.7
```

```
UPPER-BOUND ON OPTIMUM:    8980.1
PAYOFF:    8970.2      RESOURCES LEFT:    2.0

DO THESE:
      2
      3
      4
      6
      7
     13
     16
     17
     18
     19
     20
     24
     25
     27
     28
     33
     41
     42
     44
     45
     46
     47
     48
     50
     51
     52
     53
     58
     60
     61
     62
     64
     65
     66
     67
     71
     72
     73
     74

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:        0
SLIPPING:    117

RIGHT-HAND-SIDE:    4680.0
UPPER-BOUND ON OPTIMUM:    9972.6
PAYOFF:    9959.0      RESOURCES LEFT:    4.0

DO THESE:
      2
      3
      4
```

6
7
12
13
16
17
18
19
20
24
27
28
33
41
42
44
45
46
47
48
50
51
52
53
57
58
60
61
62
64
65
66
67
71
72
73
74

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 2
INITIAL SOLUTION: 0
CLIPPING: 100

RIGHT-HAND-SIDE: 4700.3
UPPER-BOUND OF OPTIMUM: 9165.1
PAYOFF: 5169. RESOURCES LEFT: 5.0

0 THESE:
2
3
4
6
7
12
13
16
17

18
19
20
24
25
27
28
33
41
42
44
45
46
47
48
50
51
52
53
57
58
60
61
62
64
65
66
67
71
72
73
74
75

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 0
SLIPPING: 100

RIGHT-HAND-SIDE: 4800.0
UPPER-BOUND ON OPTIMUM: 9257.3
PAYOFF: 9242.3 RESOURCES LEFT: 5.0

DO THESE:
2
4
6
7
8
13
16
17
18
19
20
24
27

28
33
41
42
43
44
45
46
47
48
50
51
52
53
58
60
61
62
64
65
66
67
71
72
73
74

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 2
SLIPPING: 100

RIGHT-HAND-SIDE: 4900.0
UPPER-BOUND ON OPTIMUM: 9348.2
PAYOFF: 9340.0 RESOURCES LEFT: 2.0

DO THESE:
2
4
6
7
8
12
13
16
17
18
19
20
24
25
27
28
33
41
42
44

```
      45
      46
      47
      48
      50
      51
      52
      53
      58
      60
      61
      62
      64
      65
      66
      67
      71
      72
      73
      74

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:     0
INITIAL SOLUTION:     0
SLIPPING:    100

RIGHT-HAND-SIDE:    5000.0
UPPER-BOUND ON OPTIMUM:    9439.1
PAYOFF:    9431.2     RESOURCES LEFT:    23.0

DO THESE:
       2
       4
       6
       7
       8
      12
      13
      16
      17
      18
      19
      22
      24
      25
      27
      28
      33
      41
      42
      43
      44
      45
      46
      47
      48
      51
```

```
51
52
53
57
58
59
60
61
62
64
65
66
67
71
72
73
74

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES:      2
INITIAL SOLUTION:      0
SLIPPING:    100


RIGHT-HAND-SIDE:    6500.0
UPPER-BOUND ON OPTIMUM:   10601.8
PAYOFF:  10506.0      RESOURCES LEFT:     9.0

DO THESE:
     1
     2
     3
     4
     6
     7
     8
    12
    13
    16
    17
    18
    19
    20
    24
    25
    26
    27
    28
    29
    33
    36
    39
    41
    42
    43
    44
    45
    46
```

47
48
49
50
51
52
53
54
57
58
59
60
61
62
64
65
66
67
68
70
71
72
73
74

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 17
SLIPPING: 66

RIGHT-HAND-SIDE: 8000.0
UPPER-BOUND OF OPTIMUM: 11563.9
PAYOFF: 11556. RESOURCES LEFT: 9.0

DO THESE:
1
2
3
4
5
6
7

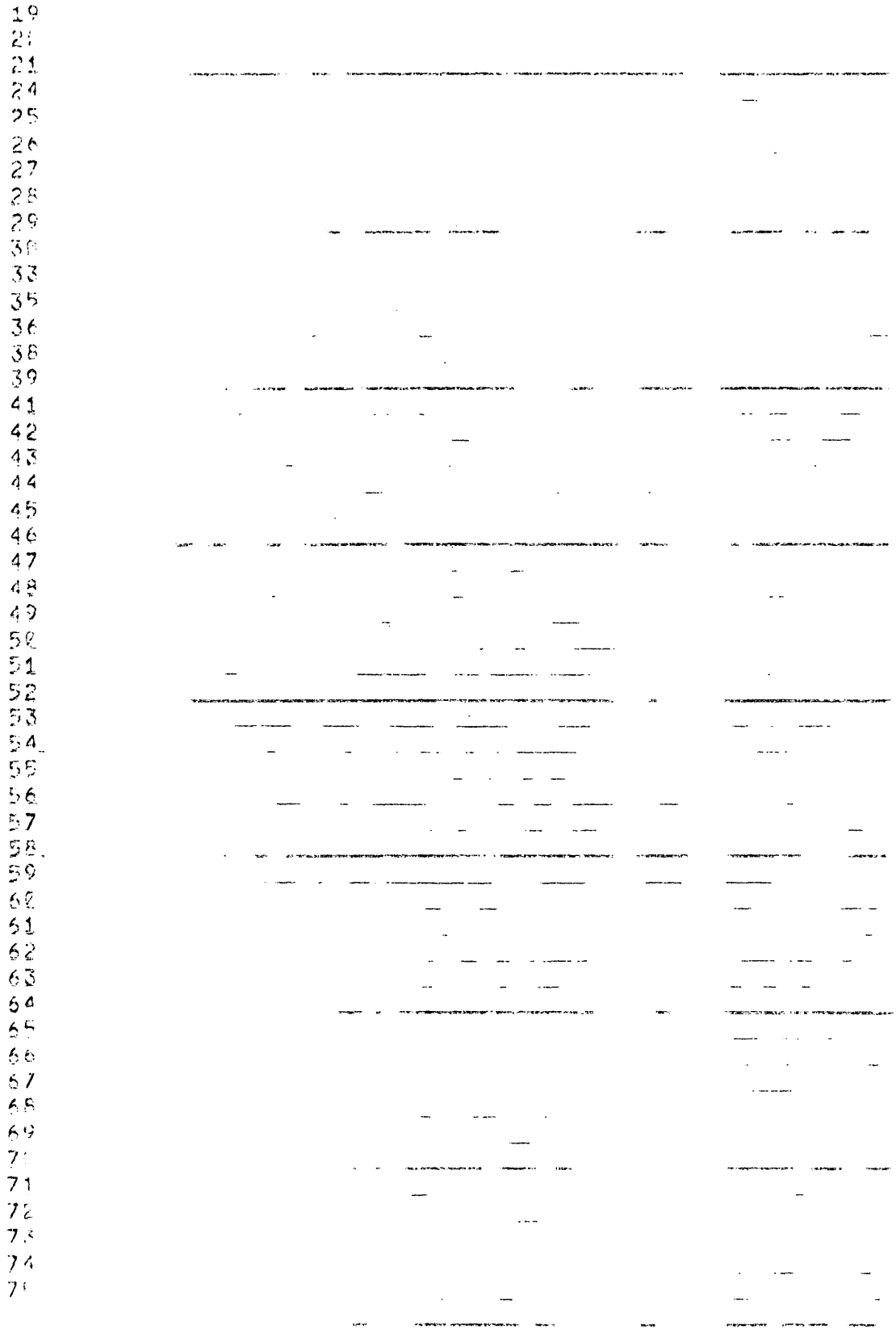
12
13
16
17
1
19
21
24
25
26
27
28
29
30

33
34
39
41
42
44
45
46
47
48
49
50
51
52
53
54
57
58
59
60
61
62
64
65
66
67
68
69
71
72
73
74

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 0
INITIAL SOLUTION: 0
SLIPPING: 84

RIGHT-HAND-SIDE: 1[illegible]000.0
UPPER-BOUND OF OPTIMUM: 1217[illegible].4
PAYOFF: 12024. RESOURCES LEFT: 13.0

TO THESE:
1
2
3
4
5
6
7
8
1[illegible]
12
13
16
17
18



I [illegible] SUMMARY, ILLISE [illegible] :
[illegible] VARIABLES: 0
[illegible] OBSERVATIONS: 0
[illegible] 5%

RIGHT-HAND-SIDE: 12[illegible]

UPPER-BOUND OF OPTIMUM: 12616.1
PAYOFF: 12224. RESOURCES LEFT: 194.0

TO THESE:

1
2
3
4
5
6
7
8
9
10
12
13
15
16
17
18
19
20
21
24
25
26
27
28
29
30
32
33
35
36
38
39
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64

65
66
67
68
69
70
71
72
73
74
75

TIMING SUMMARY, MILLISECONDS:
ORDERING VARIABLES: 3
INITIAL SOLUTION: 0
SLIPPING: 34

RIGHT-HAND-SIDE: 14000.0
UPPER-BOUND ON OPTIMUM: 12895.2
PAYOFF: 12878 RESOURCES LEFT: 37.0

DO THESE:
1
2
3
4
5
6
7
8
10
11
12
13
16
17
18
19
20
21
24
25
26
27
28
29
31
32
33
35
36
38
39
41
42
43
44

```
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
```

```
TIMING SUMMARY: MILLISECONDS:
ORDERING VARIABLES:      0
INITIAL SOLUTION:        0
SLIPPING:       33
```

BIBLIOGRAPHY

Brooks, R. B., and Geoffrion, A. M., 1966, Finding Everett's LaGrange multipliers by linear programming: Operations Research, v. 14, p. 1149.

Dantzig, G. B., 1957, Discrete-variable extremum problems: Operations Research, v. 11, p. 399.

Dorn, W. S., 1971, On LaGrange multipliers and inequalities: Operations Research, v. 19, p. 95.

Fox, B. L., and Landi, D. M., 1970, Searching for the multiplier in one-constraint optimization problems: Operations Research, v. 18, p. 253.

Geoffrion, A. M., 1969, An improved implicit enumeration approach for integer programming: Operations Research, v. 17, p. 437-454.

_____ 1970, Integer programming by implicit enumeration and Balas' method: SIAM Review, v. 9, no. 2, p. 178-190.

Gilmore, P. C., and Gomory, R. E., 1963, A linear programming approach to the cutting stock problem - Part II: Operations Research, v. 11, p. 863.

_____ 1965, Multistage cutting stock problem of two and more dimensions: Operations Research, v. 13, p. 94.

_____ 1966, The theory and computation of knapsack functions: Operations Research, v. 14, p. 1045.

Gomory, R. E., 1965, On the relation between integer and non-integer solutions to linear programs: Proc. Nat. Acad. Sci., v. 53, p. 260-265.

Greenberg, Harold, and Hegerich, R. L., 1970, A branch search algorithm for the knapsack problem: Management Sci., v. 16, p. 327.

Kaplan, Seymour, 1966, Solution of the Lorie-Savage and similar integer programming problems by the generalized LaGrange multiplier method: Operations Research, v. 14, p. 1130.

Kolesar, P. J., 1967, A branch and bound algorithm for the knapsack problem: Management Sci., v. 13, no. 9, p. 723-735.

Lawler, E. L., and Bell, M. D., 1966, A method for solving discrete optimization problems: Operations Research, v. 14, p. 1098.

Nemhauser, G. L., and Ullman, Zov, 1968, A note on the generalized LaGrange multiplier solution to an integer programming problem: Operations Research, v. 16, p. 450.

Pandit, Nahari, S. N., 1962, The loading problem: Operations Research, v. 10, p. 639.

Plateau, G., and Fayard, Didier, 1971, Contribution à la résolution du problème du knapsack: étude comparative de méthods: Docteur de Specialite thesis, Universite de Lille, France.

Pugh, G. E., 1964, LaGrange multipliers and the optimal allocation of defense resources: Operations Research, v. 12, p. 543.

Senju, Shizuo, and Toyoda, Yoshiaki, 1968, An approach to linear programming with 0-1 variables: Management Sci., v. 15, p. B-196.

Shapiro, J. F., and Wagner, H. M., 1967, A finite renewal algorithm for the knapsack and turnpike models: Operations Research, v. 15, p. 319.

Trauth, C. A., Jr., and Woolsey, R. E., 1968, Integer linear programming: a study in computational efficiency: Management Sci., v. 15, p. 481.