# EVALUATION OF A DEPTH-FIELD SEARCH ALGORITHM FOR CRITICAL PATH METHOD

By

JEFFREY A. SCOFIELD

Bachelor of Science

Oklahoma State University

December 1982

Submitted to the Faculty of the Graduate College
of the Oklahoma State University
in partial fulfillment of the requirements
for the Degree of
MASTER OF BUSINESS ADMINISTRATION
July, 1984

EVALUATION OF A DEPTH-FIRST SEARCH

ALGORITHM FOR CRITICAL

PATH METHOD

Report Approved:

Mitchell O. Locks

Adviser

Director of Graduate Studies

Head, Department of Management

ACKNOWLEDGEMENTS

I would like to express my deepest appreciation to Professor Mitchell Locks, who provided me with support and advice on this report and to Professor Marilyn Kletke, who assisted me with analyzing the computer programs.

In addition, I would like to thank Professor Gerald Oberlander in the College of Engineering and Bruce Lee of Phillips Petroleum, Inc., for providing me with real industry project planning problems. Finally, my sincerest thanks to a friend, Irene Larson, who assisted me with the formatting, editing, and typing of this paper.

TABLE OF CONTENTS

Page

## LIST OF TABLES



## LIST OF FIGURES

Name: Jeffrey A. Scofield Date of Degree: July, 1984

Institution: Oklahoma State University

Location: Stillwater, Oklahoma

Title of Study: EVALUATION OF A DEPTH-FIRST SEARCH ALGORITHM FOR CRITICAL PATH METHOD

Pages in Study: 85 Candidate for Degree of Master of Business Administration

Major Field: Business Administration


Scope and Method of Study: This paper is concerned with the testing and evaluating of a computer software program developed for a new algorithm using depth-first search processes for determining the critical path of a project planning or scheduling network.

Findings and Conclusions: The DFS-CPM program was further tested and evaluated through the use of real industry project planning and scheduling problems. Results indicated that the amount of memory required for DFS is contingent upon the structural configuration of the network as well as the number of activities. However, the execution time appeared to increase linearly with the number of activities.

Evaluations suggest that DFS-CPM possesses a promising future in the area of sensitivity analysis.


Adviser's Approval Mitchell O. Locke

# 100. INTRODUCTION

The purpose of this paper is to test and evaluate a computer program written by S. Srinivason [13] on a new algorithm developed by M. O. Locks [8] for finding the critical path of a network. In accomplishing this objective a literature review on pertinent topics will be performed and summarized. Both the CPM conventional method and DFS CPM will be reviewed. Finally, actual industry CPM problems will be solved using both methods and comparisons and evaluations will be made.

## 110. Background of Network Analysis Techniques

Project-network analysis techniques have proved to be a successful tool in effective project planning, scheduling, and control. The bar chart as developed by Gantt [10] was the primary technique available until the fifties. In 1958 two new methods of project-network analysis were developed. These are: critical path method and project evaluation and review technique (PERT).

CPM and PERT are similar in several aspects. However, CPM requires that the time elements be deterministic in nature and is used largely by management in resource allocations while PERT is used with networks probabilistic in nature.

### 111. Depth-First Search

Depth-first search is a graph theory introduced in 1972 and Targan and Hopcroft [15]. In implementation of DFS Targan [15] used a backtracking technique as described by Phillips [10].

The search in backtracking proceeds in a sequential manner. Each time the search reaches an end state, it backs up to the previous investigated state and continues the forward search from that point. In DFS the search proceeds from the source to the terminal, backing up only once the terminal has been reached.

### 120. Literature Review

Although the intent of a literature review was to find case studies of CPM being used on actual, large scale problems, several related studies on network analysis were found and will be summarized.

### 121. New Techniques in Network Analysis

Arthur J. Nevins [9] examined the use of best bud search, a general purpose heuristic program, in application to assembly line balancing. This search procedure bears some relationship to that of branch and bound.

The effectiveness of a search procedure can depend critically upon the way it (1) evaluates the relative merits of alternative paths, and (2) reacts to the disappointment of learning that a path is no longer as promising as it had previously seemed. It was in this second aspect that the best bud technique proved to be superior. However, it was also noted that while an advantage of branch and bound not shown by the best bud method is that a solution obtained using branch and bound will always be optional. Nevins concludes that the results indicate that the heuristic program is quite effective in its role as an assembly line balancer.

A paper by Samli and Bellas [1] investigate the use of graphical evaluation and review technique (GERT) in the planning and control of marketing research. They contend that GERT overcomes many PERT deficiences which makes it valuable in planning and control. In PERT, branching from a node is deterministic, since every branch (activity) must be completed at some time. Whereas GERT permits probabilistic branching and therefore, a given activity may not be undertaken under certain conditions or with a certain probability. Individual activities are assigned a probability of occurrence, a parameter set for time, and a distribution type.

Steinberg and Napier [14] developed an optimal procedure for the multi-period, multi-product, multi-level lot sizing problem by modeling the network system as a constrained generalized network. This procedure enabled the relaxation of some of the more restrictive assumptions of previous models such as those designed for production structures with single sources and/or successors.

Nicolai Siemens [12] introduced an algorithm for efficiently shortening the duration of a project when the expected project duration exceeds a predetermined limit. This algorithm is claimed to be considerably less complex than the analytic methods currently available. Hence, this method proves to be a valuable tool for managers with a CPM time-cost tradeoff problem who do not have access to a computer.

## 11. Resource Allocations

A paper by John M. Burt [4] addresses the problem of planning and controlling (through resource allocation decions) projects under conditions of uncertainty. Particular attention was given to the dynamic reallocation of resources over the duration of projects as information about actual activity times become known. This paper focused on the fact that decision making is not a static concept. That is, over the duration of a project, the project manager sequentially reallocates resources in light of new information on the status of the project. The problem of reallocation is further complemented upon by Hastings [5]. His research incorporated a branch and bound algorithm.

## 123. Additional Network Analysis

J. F. Raimond [11] applied the direct search scheme to the problem of finding a minimaxinal path in a disjunctive PERT network. By use of an implicit enumerative scheme he formulated the problem as a mixed-integer linear program. This procedure yielded an optimal solution for a small scale problem and with very few changes it can find suboptiomal solutions for larger problems.

In another study, Zangwill [16] analyzed the multi-echelon system. It was shown that the multi-echelong structure can be represented as a single source network and can thereby be analyzed by applying the theory of concave cost networks, also described by Zangwill in a previous paper.

## 124. Biasing in PERT

Theory and Monte Carlo simulation have shown that the PERT method yields results which are biased high. Klinger [7] discusses a real example involving condition under which the bias was very large. The underlying purpose of his paper was to show that the manager can be grossly misled into thinking that his chances in project completion are very good, when in reality they are very poor.

## 125. Accomplishments in DFS

In his most recent published paper on DFS Berztiss [2] developed the atomic approach for storage of information about the network. This atomic approach consisted of breaking up the DFS tree into atomic work units and to store network information into node and arc tables. Specific information pertaining to each activity on the tree is stored in the arc table. A set of data is generated and stored in the node table which views each event as the starting event of an activity. A k-tree representative of the network is constructed from the atomic trees and the arc and node tables are used to obtain the data storage structure in terms of a final node and arc table.

Berztiss [2] noted several advantages of the use of depth-first k-trees. First, was its capability of using standard tree transversal algorithms. Another advantage was the ability to write the algorithm clearly and in a non-recursive manner. Finally, the algorithm appeared to be very user friendly in the entry of data.

## 126. Case Study

Richard Booker [3], training supervisor at Occidental Petroleum Company, performed a small scale, functional, case study on PERT. His intentions were to illustrate the basic steps required in applying network techniques rather than evaluation of a particular technique or algorithm. Four basic steps were applied to the scheduling of an internal sales seminar. The first step was to estimate all the activities in the project and their corresponding durations. The second step in PERT planning is to determine the time relationships among the activities. The third step is to draw the network consisting of events and activities. The fourth and final step is to identify the critical path.

The literature review as presented only serves to be representative of the work performed in relation to network analysis and should indicate some of the faucets of the subject currently being investigated. Upon completion of this review no published case studies of CPM usage on large-scale, non-fabricated problems were found.

## 200. CRITICAL PATH METHOD

An important consideration in project scheduling or planning is the minimum time required to complete the project. CPM prvides a means of finding the longest path, or sequence of connected activities through the network, which is identified as the critical path in that it determines the duration of the project. That is, the cumulative times of the sequential activities along the critical path yield the project's length or duration. Various CPM algorithms have been introduced, most notably is the conventional method.

### The Conventional Method

Several sources of literature and textbooks describe the CPM conventional method in detail, for example see [10]. The critical path calculations include two phases. The first phase is the forward pass where calculations begin from the start node and move to the end node. At each node the earliest occurrence time of the corresponding event (ES) is calculated. The earliest starting time of one project activity is the latest possible early finish (EF) of its predecessors. An activity, EF is its ES plus the duration time of that activity (te), EF = EStte.

The second phase is the backward pass which begins calculations from the end node and moves back toward the start node, representing the latest occurrence time of the corresponding event (LF). The LF of an event is the earliest late start of all its successors, LS = LF - te.

Upon completion of forward and backward pass computations, the critical path or paths can be determined. After the critical path has been determined, the floats for the non-critical activities must be computed. All critical activities will have zero float/slack values. There are two important types of float associated with the conventional CP method: total float and free float. The total float (TF) for an activity is the difference between the maximum time available to perform the activity and its duration; that is, TF = LF - ES - te. The free float (FF) is defined by assuming that all the activities start as early as possible. Thus the free float for an activity is the excess of available time over its duration; that is, $FF_{ij}$ = ESj - ESi - teij.

In the conventional critical path method both a forward and backward pass must be performed on a project network to determine the critical path. Depth-First-search (dfs) CPM possesses a unique feature in that the critical path can be determined from only one pass.

## 220. Depth-First Search Critical Path Method

Depth-First Search is a form of sequencing and subsequently backtracking for determination of the critical path and the floats of a project scheduling network. Implementation of CPM-dfs involves two distinct steps: first, build the tree from the scheduling network, and second, perform the necessary calculations to determine the critical path or paths and the float times.

The terminology and the form of the dfs search tree used and their relation to the parent CP network follows that described by Tarjan [15].

## 230. Description of DFS-CPM

In CPM-dfs the performance of both the forward pass and backward pass is not necessary, as in the conventional CP method to determine the critical path(s). However, for slack calculations both forward and backward passes are required.

### 231. Forward Pass

The forward pass or forward search is initiated at the left-hand side of the parent tree or the start node. Each link of the dfs tree denotes a subpath originating at the start made with an activity. In dfs tree building a link can consist of several sequential activities, of the parent graph, which contain no intermediate incoming or outgoing vertices and edges.

The descriptions and rules pertaining to CPM-dfs use terms which emphasize the family relationships between the elements of the search tree. The left most node of any two nodes connected by a given link is the father and the right node is the child or son. When a node has two or more children, they are called brothers. One brother is designated as the eldest brother and is in turn processed before a younger brother.

There are three basic rules which pertain to DFS. The first is that a father is always processed before his child. Secondly, an elder brother is processed before his younger brother. Finally, all nodes which are descendants of an ancestor will be processed prior to any younger brothers of this ancestor node. Although not necessarily the critical path, a path from the start node to the end or terminal node is found at the outset. As prescribed under these rules DFS begins at the start node and proceeds all the way to the end of the graph or terminal before backtracking, that is, we first determine the depth of the graph in our search. Then the length of every path and subpath is measured against the initial path as a standard. By following these procedures CPM-DFS not only finds the critical path (longest duration), but also all of the longest subpaths to every event vertex.

### 232. Example of Forward Search

In Figure 1, the graph has 8 events and 13 activities. The activity duration times are given in parentheses.

The search tree for example 1 is given in Figure 2 and the results in Table 1. The critical path is 1 - 3, 3 - 2, 2 - 5, 5 - 8 with a total duration time of 25. For the search serial sequences 4 - 7, 7 - 8 are processed as if they were single activities. The tree's links are labeled [a, d] where a is the activity and d is the activity duration.

Table I gives for each node in Figure 2 the last CPM-vertex in the subpath, and the maximum subpath times. For subpaths having different times only the longest time is displayed. For example, for node 8 corresponding to vertex 6 the successive sequence is:

(1 - 3, 3 - 6): 11
(1 - 4, 4 - 6): 7

Subpath (1 - 3, 3 - 5, 5 - 8) with a sum activity duration of 19 is shorter than the previous subpath of ( 1 - 3, 3 - 2, 2 - 5, 5 - 8) through vertex 5, thus a backtrack is performed. Figure 2 shows that the forward search starts at source s and ends at intermediate vertices. The backward pass is simply the dual of the forward pass.

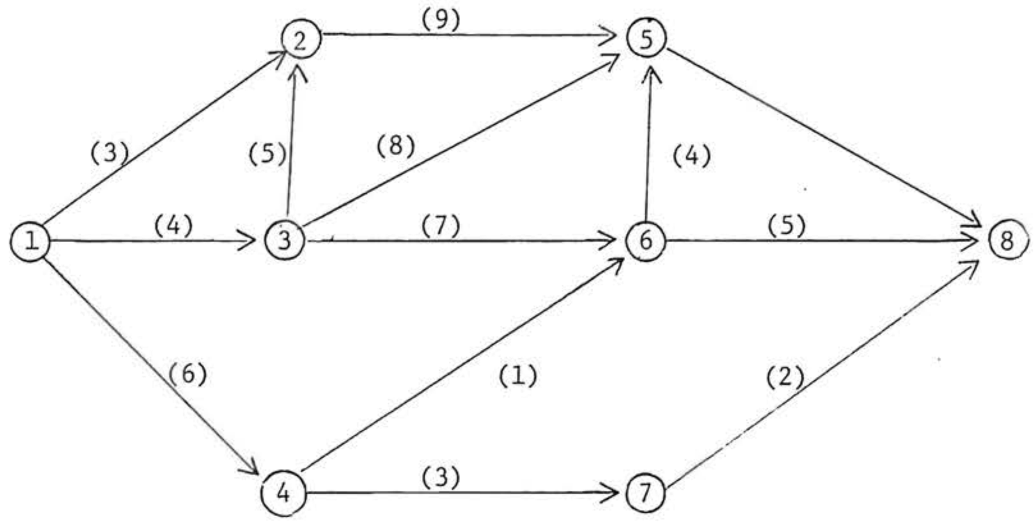



Fig. 1. CPM Graph for example 1

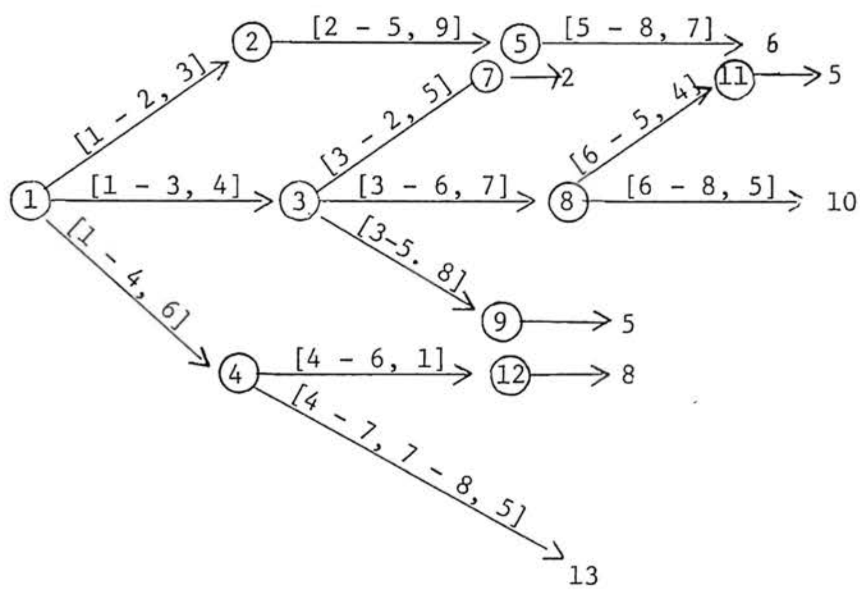



Fig. 2. Forward Search Tree for example 1.

TABLE I

ANALYSIS OF THE FORWARD SEARCH TREE IN FIGURE 2

| Node | Last v in subpath | Maximum subpath times |
|---|---|---|
| 1 | 1 | 0 |
| 2 | 2 | 9 |
| 5 | 5 | 18 |
| 6 | 8 | 25 |
| 3 | 3 | 4 |
| 7 | 2 | 9 |
| 8 | 6 | 11 |
| 10 | 8 | 16 |
| 9 | 5 | 12 |
| 4 | 4 | 6 |
| 11 | 5 | 15 |
| 12 | 6 | 7 |
| 13 | 8 | 11 |

240. Backward Pass

The search tree for the backward pass of the CPM network in Figure 1 is given in Figure 3 and the analysis is provided in Table II.

250. Floats

In implementation of DFS-CPM five types of activity float times will be examined. These include: total float, free float, independent float, safety float and event float.

Total float (TFij) for activity (ij) is the difference between the maximum time available to perform the activity ($LC_j$ - ESi) and its duration (Tij); that is, TFij = LFj - ESi - Tij. The free float is defined by assuming that all the activities start as early as possible. The free float (FFij) for activity (ij) is the excess of available times (ESj - ESi) over its duration; that is, FFij = ESj - ESi - Tij. The independent float of a particular activity is the maximum time that that activity can be delayed without delaying successor activities, if all prior activities are finished as late as possible. That is, it provides a measure of time available given worse case conditions of predecessor activcities. It is calculated by
IFij = TFij - [LFi +Tij]. Safety float is the maximum amount of time an activity can be delayed without affecting the final project completion if all predecessor activities are completed as late as possible. Hence, SFij = LFij - [LFj + Tij].

The concept of event float, as presented by M. O. Locks, consists of extending the basic idea of float to events. The event float is defined as the amount of slack time available at a particular event

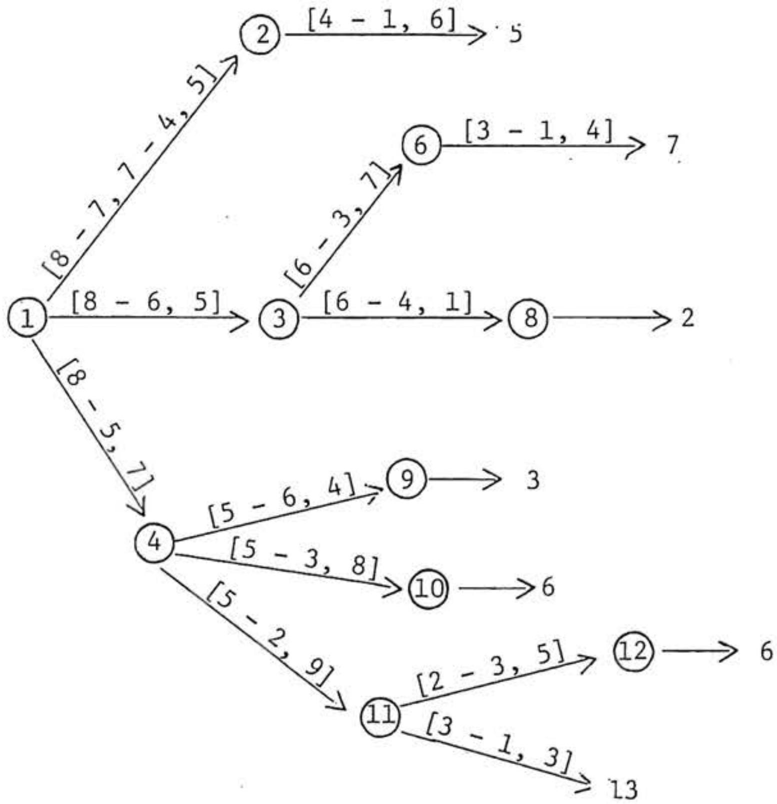



Fig. 3. Search Tree of Backward Pass for example 1.

TABLE II

ANALYSIS OF THE BACKWARD SEARCH TREE IN FIGURE 3

| Node | First v in subpath | Maximum subpath times |
|---|---|---|
| 1 | 8 | 0 |
| 2 | 4 | 12 |
| 5 | 1 | 18 |
| 3 | 6 | 11 |
| 6 | 3 | 21 |
| 7 | 1 | 25 |
| 8 | 4 | 12 |
| 4 | 5 | 7 |
| 9 | 6 | 11 |
| 10 | 3 | 15 |
| 11 | 2 | 16 |
| 12 | 3 | 21 |
| 13 | 1 | 19 |

when all preceding and succeeding activities are carried out in the shortest possible time while the total project time remains unaffected.

Any activity or event on the critical path will have float times of 0 for each of the five types of floats previously discussed.

## 300. PRACTICAL INDUSTRY EXAMPLES

Three individual, real industry CPM problems were used to test and evaluate the computer program for the algorithm developed by M. O. Locks on DFS-CPM [8]. The first is a small scale problem taken from Booker [3]. The second was a problem provided by Phillips Petroleum, Inc. The third problem was of the scheduling of the construction of a major model provided by Prof. Gerald D. Oberlender who acted as a consultant on the project.

### 310. Booker's Sales Seminar

Richard Booker, coordinator of marketing education, had the responsibility for planning and implementing a 7-day international sales seminar. Management desired completion of preseminar activities within 22 working days.

This CPM project was solved using both the DFS computer software and with the conventional computer method program.

The results from the conventional displayed the critical path as found in Booker's paper along with the float times. The critical path was 1 - 2 - 3 - 4 - 5 - 6 - 10 - 13 - 14 - 15 with a duration of 21 days. The program required .12 seconds of compilation time. This is the amount of time required by the computer to compile the program and is not effected in any means by the amount of data present. The actual time required to execute the program (execution time) was .03 seconds or

30 millisconds. The total amount of memory used for storage (total storage) was 25K (25,000) bytes (Appendix 501).

The DFS program not only produced the critical path of 1 - 2 - 3 - 4 - 5 - 6 - 10 - 13 - 14 - 15 but also revealed a dual critical path of 1 - 2 - 3 - 4 - 5 - 8 - 10 - 12 - 13 - 14 - 15 again with a critical time of 21 days. The compilation time was .17 seconds and the execution time was .05 seconds. The total storage was 30K bytes (Appendix 502).

Thus from this CPM project it is observed that the DFS required 20 milliseconds longer to execute along with 5000 bytes more total storage than the conventional method.

## 320. Phillips Project

In October 1983 Phillips Petroleum Inc. implemented Rock Creek Turnaround project. Phillips scheduled the project using a CPM software package produced by Premis. This project consisted of 150 activities along with several dummy activities. Rock Creek Turnaround was the renovation of an oil refinery where actual down time was very costly and desired to be kept at a minimum.

The DFS located a critical path consistiong of 11 sequential events with a sum duration of 256 hours. The compilation time was .16 seconds and the execution time was .38 seconds. The DFS program required 120K bytes of total storage (Appendix 503).

The critical path, critical time, and float times were identical for the conventional method (Appendix 504). However, the compilation time was .12 seconds and the execution time was .33 seconds. The total amount of storage used was 39K bytes.

230. Motel Construction Project

Due to the request of the construction company and G. D. Oberlender the name of the motel will not be disclosed. The CPM Project was that of the construction of a large motel, the motel being one of a major chain across the United States. The network covered the entire construction of the project. The network consisted of 200 activities which resulted in an arrow presidence diagram consisting of 254 events (Appendix 505).

The CPM conventional algoritym solved the network with an execution time of .83 seconds and required a total memory storage of 57K bytes.

Solution by DFS algorithm identified two parallel critical paths. Execution time was .67 seconds and total storage was 109K bytes (Appendix 506).

# 400. EVALUATION

This section will deal with addressing the results of section 300 in terms of comparison of DFS and the conventional CPM methods. In addition assessments will be made in regard to using the DFS-CPM program as an additional software package.

## 410. DFS as an Operational Program

### 411. Entry of Data

Current means of data entry requires the user to edit the program each time data is entered or time restrictions are changed. The program could be made more user friendly by prompting the user for data entry or changes upon execution rather than requiring editing.

Data entry requires the presence of an arrow diagram, (i.e., circles as nodes interconnected by arrows as activities). Network diagrams are not always readily available for extremely large-scale problems. In addition, most industry project networks are in the form of activity precedence diagrams (APD), both the Phillips and the construction projects were APD, which makes it very difficult and time consuming to transfer the APD to an arrow diagram. Thus, for practical use it would be advantageous to change the structure of the program so data entry can be made directly from an APD.

### 412. Work Codes

Several activities or tasks in a project are subject to time and worker restrictions. For example, the paving or blacktopping of a drive or parking lot must be done in daylight. Various activities subject to time restrictions could be coded and the computer program could evaluate these activities accordingly.

### 413. Updating Activities

We can distinguish three types of updating activities to which a scheduling network may be subjected.. These include: 1) changes to durations alone, 2) changes to the structure of the network, and 3) changes to structure and durations.

Typically as projects are updated the entire network is reconstructed from source to terminal. However, activities which have already been completed are no longer of grave importance and really need not be shown. For example Figures 4 and 5.

At time A all previous activities, i.e., nodes 1-4 are of no concern. The time remaining for activities A, B, and C along with all successive activities are important to the manager. Hence, a program which could update the project in this manner could prove to be very useful.

### 414. Data Output

Data output of the critical path, critical time, and various floats would be very confusing to a project manager. A manager without expertise in CPM would likely find it easier to interpret early start, early finish, late start and late finish.

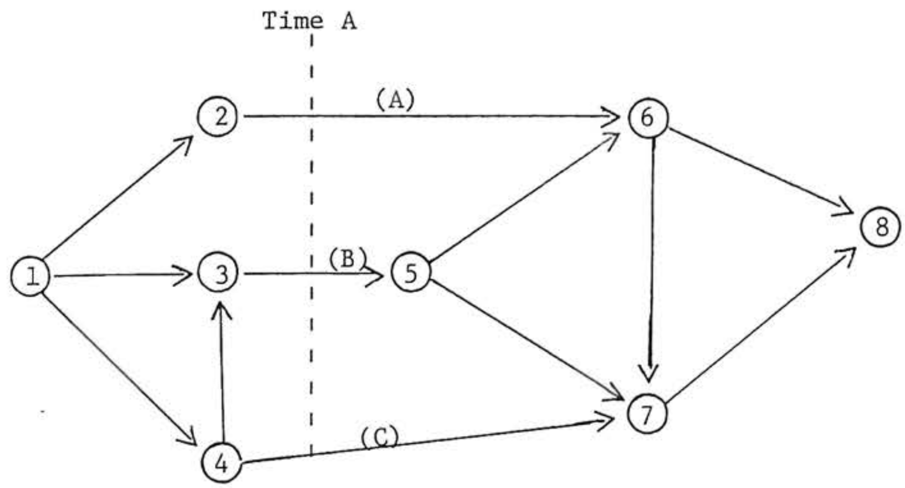



Fig. 4. Example Network

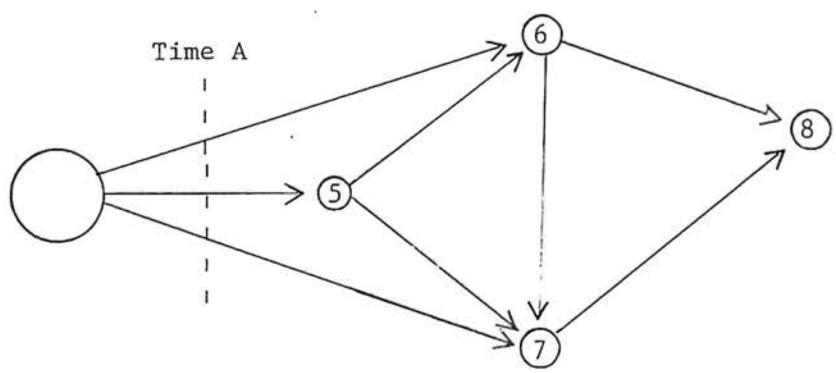



Fig. 5. Updated Network

It should be noted that these are casual observations regarding the use of DFS-CPM as an operational program. However, DFS-CPM was developed as a means of further investigating the use of depth first in application to the critical path method and not as a complete operational package.

## 420. Computer Memory Usage and Executive Time

Both the DFS and conventional programs were run with only one activity with 0 duration. This was to evaluate the compilation time and memory usage with essentially no date. The compilation time for DFS is .15 seconds and the total storage is 24K bytes (Apppendix 507) as compared to a .12 compilation time and 23K bytes of total storage for the conventional CPM program (Appendix 508). These differences are not all that unusual in consideration that the CPM-DFS program had 468 cards or 244 executable statements as compared to 401 cards or 216 executable statements in the conventional CPM program.

In the conventional algorithm the execution time and the total storage increased exponentially in respct to the number of activities with the execution time increasing at a much greater extent (Appendices 509 and 510).

The execution time for the DFS algorithm increased linearly with respect to the number of activities (Appendix 511). Execution time for DFS was greater than that for the conventional algorithm up to approximately 175 activities at which the execution time for the conventional program increased sharply beyond that for DFS.

Total memory storage for DFS increased at a fairly linear rate up to 150 activities but reached a peak and then decreased for the motel construction project (Appendix 512). In analyzation of this phenomenon trace statements were inserted into the DFS program. These trace statements indicated that the Phillips program recalled the build-tree recursive subroutine more often than the construction project. This is easily explained by the fact that the Phillips network was much greater in breadth than the constuction project. That is, a given node on the Phillips network was likely to have several branches whereas the nodes on the network for the construction project usually only consisted of one or two branches.

## 430. Program Modifications

Two changes were made to the DFS-CPM program. These include a change in the output of data and internal memory allocation.

### 431. Data Output

In its original form the program will have to list all executable and comment statements along with the output data when executed. This was very time consuming and unnecessary. To alleviate this problem a nosource command was inserted with line 50 of the JOL Control Language. Now only the critical path, critical time, and float times are displayed upon execution. However, for error detection this statement should be removed.

The second change consisted of increasing internal memory allocations. Although this program was previously tested with a large-scale sample, the sample was small in girth. That is, although it had several sequential activities the span from any one node was 2 to 4 activities.

The Phillips problem had 20 activities spanning from a single predecessor.

## 440. Concluding Comments

Having effectively tested the DFS software program with real project planning problems one can access that DFS does have a promising futre in critical path method.

From the investigation performed it is evident that the memory used in DFS is not only contingent upon the size of the network but also its structural configuration. A possible explanation of the large differences in total storage for the conventional and DFS programs could be that the conventional algorithm lacks the capability of determining the existence of parallel paths. This feature of DFS alone required additional executable statements which in turn would require more memory upon execution.

In theory DFS algorithm does not require dummy activities. However, for this computer program dummy activities must be inserted wherever two parallel events exist. This fact hampers a complete and successful evaluation of a depth-first process for the critical path method.

It is the conclusion of this author that DFS possesses unique characteristics which could make it useful in various forms of sensitivity analysis, such as described in section 413, and suggest further investigation of depth-first search.

## BIBLIOGRAPHY


[1] Bellas, Carl and Samli, A. Coskun, "The Use of Gert in the Planning and Control of Marketing Research," Journal of Marketing Research 8, (August 1979): 335-9.

[2] Berztiss, A. T., "Depth-First K-trees and Critical Path Analysis," Acta Information. 13, (1980): 325-46.

[3] Booker, Richard, " A Pert Case Study", Training in Business and Industry, (April 1974): 41-2.

[4] Burt, John M., "Planning and Dynamic Control of Projects Under Uncertainty", Management Science 24, (November 1977): 249-58.

[5] Hastings, N. A. S., ""On Resource Allocation in Project Networks," Operational Research Quarterly 23, (June 1972): 217-21.

[6] Kelley, Albert J., New Dimensions of Project Management Lexington, MA>: Lexington Books, 1981.

[7] Klingel, A. R., "Bias in Pert Project Completion Time Calculations for a Real Network", Management Science 13, (December 1976): 194-201.

[8] Locks, M. O., "Optimal Search for Critical Path", Unpublished Paper, Oklahoma State University, 1983.

[9] Nevins, Arthur J., "Assembly Line Balancing Using Best Bud Search", Management Science 18, (May 1972): 529-39.

[10] Phillips, Don T. and Garcia-Diaz, Alberto, Fundamentals of Network Analysis Englewood Cliffs, N. J.: Prentice-Hall, Inc., 1981.

[11] Raimond, S. F., "Maximal Paths in Disjunctive Graphs by Direct Search", IBM Journal of Research Development, (July 1969): 391-9.

[12] Siemens, N., "A Simple CPM Time-Cost Tradeoff Algorithm," Management Science, 17 (February 1971): 354-63.

[13] Srinivason, S., "An Implementation of a Single-pass Algorithm for the Critical Path Method", unpublished paper: Master's Report, (December 1983).

[14] Steinberg, Earle and Napier, H. Albert, "Optimal Multi-Level Lot Sizing for Requirements Planning Systems", Management Science 26, (December 1980): 1258-70.

[15] Tarjan, R., "Depth-first Search and Linear Graph Algorithms", Siam J. Comput., 1, (1972)" 146-60.

[16] Zangwill, Willard I., "A Backlogging Model and a Multi-Echelon Model of a Dynamic Economic Lot Size Production System--A Network Approach", Management Science 15, (May 1969): 506-27.

APPENDIX 501

Preseminar Activities, Conventional Method

PROJECT

CRITICAL PATH LENGTH = 21

| CRITICAL EVENTS | | | | | |
|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 |
| 10 | 12 | 13 | 14 | 15 | |

| EVENT | EVENT FLOAT |
|---|---|
| 1 | 0 |
| 2 | 0 |
| 3 | 0 |
| 4 | 0 |
| 5 | 0 |
| 6 | 0 |
| 7 | 2 |
| 8 | 0 |
| 9 | 2 |
| 10 | 0 |
| 11 | 2 |
| 12 | 0 |
| 13 | 0 |
| 14 | 0 |
| 15 | 0 |

PROJECT

| ACTIVITY | TOTAL FLOAT | FREE FLOAT | INDEP. FLOAT | SAFETY FLOAT |
|---|---|---|---|---|
| 1 ----- 2 | 0 | 0 | 0 | 0 |
| 1 ----- 4 | 5 | 5 | 5 | 5 |
| 2 ----- 3 | 0 | 0 | 0 | 0 |
| 4 ----- 5 | 0 | 0 | 0 | 0 |
| 5 ----- 6 | 0 | 0 | 0 | 0 |
| 5 ----- 7 | 2 | 0 | 0 | 2 |
| 5 ----- 8 | 0 | 0 | 0 | 0 |
| 6 ----- 10 | 0 | 0 | 0 | 0 |
| 7 ----- 9 | 2 | 0 | 0 | 0 |
| 9 ----- 11 | 2 | 0 | 0 | 0 |
| 10 ----- 12 | 0 | 0 | 0 | 0 |
| 12 ----- 13 | 0 | 0 | 0 | 0 |
| 13 ----- 14 | 0 | 0 | 0 | 0 |
| 14 ----- 15 | 0 | 0 | 0 | 0 |

IN STMT 10 PROGRAM RETURNS FROM MAIN PROCEDURE.

PL/C-R7 POST-MORTEM DUMP PAGE 1

IN STMT 10 SCALARS AND BLOCK-TRACE:

***** MAIN PROCEDURE CONCRIP

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| L= | ? | Z= | 2 | Y= | 5 | X= | 14 | P= | ? |
| M= | 1 | K= | 0 | J= | 15 | I= | 16 | C= | ? |
| B= | 0 | A= | 34 | TOTAL_EVENTS= | 15 | | | | |

NON-0 PROCEDURE EXECUTION COUNTS:

| NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TEST | 0189 | 00026 | CALCULATE | 0169 | 00026 | PASS | 0120 | 00002 | TRACE | 0107 | 00034 | EXECUTE | 0014 | 00001 |
| CONCRIP | 0001 | 00001 | | | | | | | | | | | | |

LABEL EXECUTION COUNTS:

| NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AHEAD | 0201 | 00026 | CONTINUE | 0186 | 00026 | FURTHER | 0182 | 00000 | HOP | 0166 | 00001 | SKIP | 0149 | 00001 |
| BACK | 0132 | 00002 | JUMP | 0105 | 00001 | NEXT | 0101 | 00017 | OUT | 0061 | 00014 | LOOP | 0033 | 00017 |

COMPILATION STATISTICS (0216 STATEMENTS)

| SECONDS | ERRORS | WARNINGS | PAGES | LINES | CARDS | INCL'S |
|---|---|---|---|---|---|---|
| .12 | 0 | 1 | 0 | 4 | 401 | 0 |

EXECUTION STATISTICS

| SECONDS | ERRORS | WARNINGS | PAGES | LINES | CARDS | INCL'S | AUX I/O |
|---|---|---|---|---|---|---|---|
| .03 | 0 | 0 | 3 | 57 | 15 | 0 | 0 |

| BYTES | SYMBOL TABLE | INTERMEDIATE CODE | OBJECT CODE | STATIC CORE | AUTOMATIC CORE | DYNAMIC CORE | TOTAL STORAGE |
|---|---|---|---|---|---|---|---|
| USED | 8199( 9K) | 5638( 6K) | 10656( 11K) | 342( 1K) | 6926( 7K) | 0( 0K) | 25342( 25K) |
| UNUSED | 33929( 33K) | 36161( 35K) | 66418( 64K) | 59142( 57K) | 59142( 57K) | 36161( 35K) | 59142( 57K) |

THIS PROGRAM MAY BE RERUN WITHOUT CHANGE IN A REGION 57K BYTES SMALLER USING TABLESIZE= 2050

# APPENDIX 502

Preseminar Activities, DFS

PROJECT

CRITICAL PATH LENGTH = 21

CRITICAL PATH

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 10 |
| 12 | 13 | 14 | 15 | 5 | 8 | 10 |
| 12 | 13 | 14 | 15 | | | |

| EVENT | EVENT FLOAT |
|---|---|
| 1 | 0 |
| 2 | 0 |
| 3 | 0 |
| 4 | 0 |
| 5 | 0 |
| 6 | 0 |
| 7 | 2 |
| 8 | 0 |
| 9 | 2 |
| 10 | 0 |
| 11 | 2 |
| 12 | 0 |
| 13 | 0 |
| 14 | 0 |
| 15 | 0 |

| ACTIVITY | TOTAL FLOAT | FREE FLOAT | INDEP. FLOAT | SAFETY FLOAT |
|---|---|---|---|---|
| 1 ----- 2 | 0 | 0 | 0 | 0 |
| 1 ----- 4 | 5 | 5 | 5 | 5 |
| 2 ----- 3 | 0 | 0 | 0 | 0 |
| 4 ----- 5 | 0 | 0 | 0 | 0 |
| 5 ----- 6 | 0 | 0 | 0 | 0 |
| 5 ----- 7 | 2 | 0 | 0 | 2 |
| 5 ----- 8 | 0 | 0 | 0 | 0 |
| 6 ----- 10 | 0 | 0 | 0 | 0 |
| 7 ----- 9 | 2 | 0 | 0 | 0 |
| 9 ----- 11 | 2 | 0 | 0 | 0 |
| 10 ----- 12 | 0 | 0 | 0 | 0 |
| 12 ----- 13 | 0 | 0 | 0 | 0 |
| 13 ----- 14 | 0 | 0 | 0 | 0 |
| 14 ----- 15 | 0 | 0 | 0 | 0 |

IN STMT 10 PROGRAM RETURNS FROM MAIN PROCEDURE.

```
                                                                     PL/C-R7   POST-MORTEM DUMP   PAGE   1

 IN STMT    10  SCALARS AND BLOCK-TRACE:

 ***** MAIN PROCEDURE DFSCRIP

 L=          0             Z=        2             Y=        4             X=       14              P=        ?
 M=          1             K=        0             J=       15             I=       16              C=       18
 B=          0             A=       34             TOTAL_EVENTS=       15


 NON-O PROCEDURE EXECUTION COUNTS:
 NAME          STMT COUNT  NAME          STMT COUNT  NAME          STMT COUNT  NAME          STMT COUNT  NAME          STMT COUNT
 CP_ROUTE      0210 00017  CRITICALPAT 0171 00036  BUILD_TREE    0134 00036  TRACE         0120 00034  SEARCH        0014 00001
 DFSCRIP       0001 00001

 LABEL EXECUTION COUNTS:
 NAME          STMT COUNT  NAME          STMT COUNT  NAME          STMT COUNT  NAME          STMT COUNT  NAME          STMT COUNT
 NEXT          0115 00017  OUT           0066 00014  JUMP          0057 00017  LOOP          0037 00017


       COMPILATION STATISTICS  (0244 STATEMENTS)     |                      EXECUTION STATISTICS
 SECONDS  ERRORS  WARNINGS  PAGES  LINES  CARDS  INCL'S | SECONDS  ERRORS  WARNINGS  PAGES  LINES  CARDS  INCL'S  AUX I/O
     .17       0         1      9    523    468       0 |     .05       0         0      3     49     15       0        0
 -------------------------------------------------------+-------------------------------------------------------------------
 BYTES    SYMBOL TABLE   INTERMEDIATE CODE   OBJECT CODE |   STATIC CORE   AUTOMATIC CORE    DYNAMIC CORE    TOTAL STORAGE
 USED      8670(   9K)        6560(   7K)    12448(  13K) |     342(   1K)     9862(  10K)        0(    0K)    30526(  30K)
 UNUSED   33458(  32K)       35239(  34K)    64172(  62K) |   53958(  52K)    53958(  52K)    35239(  34K)    53958(  52K)

 THIS PROGRAM MAY BE RERUN WITHOUT CHANGE IN A REGION  52K BYTES SMALLER USING TABLESIZE=  2168
```

APPENDIX 503

Phillips Project, DFS

CRITICAL PATH LENGTH = 256

CRITICAL PATH 1 2 4 22 43 61 62
74 80 83 118

| EVENT | EVENT FLOAT |
|---|---|
| 1 | 0 |
| 2 | 0 |
| 3 | 228 |
| 4 | 0 |
| 5 | 240 |
| 6 | 241 |
| 7 | 212 |
| 8 | 231 |
| 9 | 187 |
| 10 | 166 |
| 11 | 188 |
| 12 | 244 |
| 13 | 18 |
| 14 | 44 |
| 15 | 196 |
| 16 | 238 |
| 17 | 238 |
| 18 | 226 |
| 19 | 190 |
| 20 | 240 |
| 21 | 228 |
| 22 | 0 |
| 23 | 240 |
| 24 | 241 |
| 25 | 212 |
| 26 | 232 |
| 27 | 215 |
| 28 | 231 |
| 29 | 225 |
| 30 | 233 |
| 31 | 231 |
| 32 | 233 |
| 33 | 187 |
| 34 | 194 |
| 35 | 166 |
| 36 | 188 |
| 37 | 18 |
| 38 | 44 |
| 39 | 196 |
| 40 | 238 |
| 41 | 226 |
| 42 | 190 |
| 43 | 0 |
| 44 | 241 |
| 45 | 212 |
| 46 | 232 |
| 47 | 215 |

| | |
|---|---|
| 48 | 231 |
| 49 | 225 |
| 50 | 233 |
| 51 | 233 |
| 52 | 187 |
| 53 | 194 |
| 54 | 166 |
| 55 | 188 |
| 56 | 18 |
| 57 | 44 |
| 58 | 226 |
| 59 | 190 |
| 60 | 192 |
| 61 | 0 |
| 62 | 0 |
| 63 | 212 |
| 64 | 215 |
| 65 | 233 |
| 66 | 233 |
| 67 | 194 |
| 68 | 166 |
| 69 | 188 |
| 70 | 18 |
| 71 | 44 |
| 72 | 226 |
| 73 | 190 |
| 74 | 0 |
| 75 | 215 |
| 76 | 194 |
| 77 | 242 |
| 78 | 18 |
| 79 | 190 |
| 80 | 0 |
| 81 | 190 |
| 82 | 190 |
| 83 | 0 |
| 84 | 238 |
| 85 | 190 |
| 86 | 246 |
| 87 | 247 |
| 88 | 212 |
| 89 | 240 |
| 90 | 246 |
| 91 | 245 |
| 92 | 242 |
| 93 | 224 |
| 94 | 247 |
| 95 | 246 |
| 96 | 244 |
| 97 | 246 |
| 98 | 212 |
| 99 | 244 |
| 100 | 247 |
| 101 | 244 |
| 102 | 246 |
| 103 | 240 |
| 104 | 244 |
| 105 | 246 |
| 106 | 247 |
| 107 | 242 |

| | |
|---|---|
| 108 | 239 |
| 109 | 243 |
| 110 | 231 |
| 111 | 237 |
| 112 | 246 |
| 113 | 244 |
| 114 | 246 |
| 115 | 245 |
| 116 | 240 |
| 117 | 247 |
| 118 | 0 |

| ACTIVITY | TOTAL FLOAT | FREE FLOAT | INDEP. FLOAT | SAFETY FLOAT |
|---|---|---|---|---|
| 1 ----- 2 | 0 | 0 | 0 | 0 |
| 2 ----- 83 | 188 | 188 | 188 | 188 |
| 2 ----- 85 | 246 | 56 | 56 | 246 |
| 2 ----- 4 | 0 | 0 | 0 | 0 |
| 2 ----- 86 | 246 | 0 | 0 | 246 |
| 2 ----- 87 | 247 | 0 | 0 | 247 |
| 2 ----- 88 | 212 | 0 | 0 | 212 |
| 2 ----- 89 | 240 | 0 | 0 | 240 |
| 2 ----- 3 | 228 | 0 | 0 | 228 |
| 2 ----- 90 | 246 | 0 | 0 | 246 |
| 2 ----- 5 | 240 | 0 | 0 | 240 |
| 2 ----- 8 | 231 | 0 | 0 | 231 |
| 2 ----- 91 | 245 | 0 | 0 | 245 |
| 2 ----- 6 | 241 | 0 | 0 | 241 |
| 2 ----- 92 | 242 | 0 | 0 | 242 |
| 2 ----- 93 | 224 | 0 | 0 | 224 |
| 2 ----- 94 | 247 | 0 | 0 | 247 |
| 2 ----- 95 | 246 | 0 | 0 | 246 |
| 2 ----- 96 | 244 | 0 | 0 | 244 |
| 2 ----- 97 | 246 | 0 | 0 | 246 |
| 2 ----- 7 | 212 | 0 | 0 | 212 |
| 2 ----- 98 | 212 | 0 | 0 | 212 |
| 2 ----- 99 | 244 | 0 | 0 | 244 |
| 2 -----100 | 247 | 0 | 0 | 247 |
| 2 ----- 9 | 187 | 0 | 0 | 187 |
| 2 ----- 10 | 166 | 0 | 0 | 166 |
| 2 -----101 | 244 | 0 | 0 | 244 |
| 2 -----102 | 246 | 0 | 0 | 246 |
| 2 ----- 11 | 188 | 0 | 0 | 188 |
| 2 ----- 12 | 244 | 0 | 0 | 244 |
| 2 ----- 13 | 18 | 0 | 0 | 18 |
| 2 -----103 | 240 | 0 | 0 | 240 |
| 2 -----104 | 244 | 0 | 0 | 244 |
| 2 -----105 | 246 | 0 | 0 | 246 |
| 2 ----- 14 | 44 | 0 | 0 | 44 |
| 2 ----- 15 | 196 | 0 | 0 | 196 |
| 2 ----- 16 | 238 | 0 | 0 | 238 |
| 2 ----- 17 | 238 | 0 | 0 | 238 |
| 2 ----- 18 | 226 | 0 | 0 | 226 |
| 2 ----- 19 | 190 | 0 | 0 | 190 |
| 2 ----- 20 | 240 | 0 | 0 | 240 |
| 3 ----- 21 | 228 | 0 | 0 | 0 |
| 4 ----- 22 | 0 | 0 | 0 | 0 |
| 5 ----- 23 | 240 | 0 | 0 | 0 |
| 6 ----- 24 | 241 | 0 | 0 | 0 |
| 7 ----- 25 | 212 | 0 | 0 | 0 |
| 8 -----107 | 242 | 0 | 0 | 11 |
| 8 -----108 | 239 | 0 | 0 | 8 |
| 8 ----- 26 | 232 | 0 | 0 | 1 |
| 8 -----109 | 243 | 0 | 0 | 12 |
| 8 -----110 | 231 | 0 | 0 | 0 |
| 8 -----111 | 237 | 0 | 0 | 6 |
| 8 ----- 83 | 241 | 241 | 10 | 10 |
| 9 ----- 83 | 244 | 244 | 57 | 57 |
| 9 -----112 | 246 | 0 | 0 | 59 |
| 9 ----- 27 | 215 | 0 | 0 | 28 |
| 9 ----- 28 | 231 | 0 | 0 | 44 |
| 9 -----113 | 244 | 0 | 0 | 57 |
| 9 -----114 | 246 | 0 | 0 | 59 |

| | | | | | |
|---|---|---|---|---|---|
| 9 | ----- 29 | 225 | 0 | 0 | 38 |
| 9 | ----- 30 | 233 | 0 | 0 | 46 |
| 9 | ----- 31 | 231 | 0 | 0 | 44 |
| 9 | ----- 32 | 233 | 0 | 0 | 46 |
| 9 | -----115 | 245 | 0 | 0 | 58 |
| 9 | ----- 33 | 187 | 0 | 0 | 0 |
| 9 | -----116 | 240 | 0 | 0 | 53 |
| 9 | ----- 34 | 194 | 0 | 0 | 7 |
| 10 | ----- 35 | 166 | 0 | 0 | 0 |
| 11 | ----- 36 | 188 | 0 | 0 | 0 |
| 12 | ----- 83 | 244 | 244 | 0 | 0 |
| 13 | ----- 37 | 18 | 0 | 0 | 0 |
| 14 | ----- 38 | 44 | 0 | 0 | 0 |
| 15 | ----- 39 | 196 | 0 | 0 | 0 |
| 16 | ----- 40 | 238 | 0 | 0 | 0 |
| 17 | ----- 83 | 238 | 238 | 0 | 0 |
| 18 | ----- 41 | 226 | 0 | 0 | 0 |
| 19 | ----- 42 | 190 | 0 | 0 | 0 |
| 20 | ----- 83 | 240 | 240 | 0 | 0 |
| 21 | ----- 83 | 228 | 228 | 0 | 0 |
| 22 | ----- 43 | 0 | 0 | 0 | 0 |
| 23 | ----- 83 | 240 | 240 | 0 | 0 |
| 24 | ----- 44 | 241 | 0 | 0 | 0 |
| 25 | ----- 45 | 212 | 0 | 0 | 0 |
| 26 | ----- 46 | 232 | 0 | 0 | 0 |
| 27 | ----- 47 | 215 | 0 | 0 | 0 |
| 28 | ----- 48 | 231 | 0 | 0 | 0 |
| 29 | ----- 49 | 225 | 0 | 0 | 0 |
| 30 | ----- 50 | 233 | 0 | 0 | 0 |
| 31 | ----- 83 | 231 | 231 | 0 | 0 |
| 32 | ----- 51 | 233 | 0 | 0 | 0 |
| 33 | ----- 52 | 187 | 0 | 0 | 0 |
| 34 | ----- 53 | 194 | 0 | 0 | 0 |
| 35 | ----- 54 | 166 | 0 | 0 | 0 |
| 36 | ----- 55 | 188 | 0 | 0 | 0 |
| 37 | ----- 56 | 18 | 0 | 0 | 0 |
| 38 | ----- 57 | 44 | 0 | 0 | 0 |
| 39 | ----- 83 | 196 | 196 | 0 | 0 |
| 40 | ----- 83 | 238 | 238 | 0 | 0 |
| 41 | ----- 58 | 226 | 0 | 0 | 0 |
| 42 | ----- 59 | 190 | 0 | 0 | 0 |
| 43 | ----- 83 | 212 | 212 | 212 | 212 |
| 43 | ----- 60 | 192 | 0 | 0 | 192 |
| 43 | ----- 61 | 0 | 0 | 0 | 0 |
| 44 | ----- 83 | 241 | 241 | 0 | 0 |
| 45 | ----- 63 | 212 | 0 | 0 | 0 |
| 46 | ----- 83 | 232 | 232 | 0 | 0 |
| 47 | ----- 64 | 215 | 0 | 0 | 0 |
| 48 | ----- 83 | 231 | 231 | 0 | 0 |
| 49 | ----- 83 | 225 | 225 | 0 | 0 |
| 50 | ----- 65 | 233 | 0 | 0 | 0 |
| 51 | ----- 66 | 233 | 0 | 0 | 0 |
| 52 | ----- 83 | 187 | 187 | 0 | 0 |
| 53 | ----- 67 | 194 | 0 | 0 | 0 |
| 54 | ----- 68 | 166 | 0 | 0 | 0 |
| 55 | ----- 69 | 188 | 0 | 0 | 0 |
| 56 | ----- 70 | 18 | 0 | 0 | 0 |
| 57 | ----- 71 | 44 | 0 | 0 | 0 |
| 58 | ----- 72 | 226 | 0 | 0 | 0 |
| 59 | ----- 73 | 190 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 60 ----- 83 | 192 | 192 | 0 | 0 |
| 61 ----- 74 | 36 | 36 | 36 | 36 |
| 61 ----- 62 | 0 | 0 | 0 | 0 |
| 62 ----- 83 | 96 | 96 | 96 | 96 |
| 62 ----- 74 | 0 | 0 | 0 | 0 |
| 63 ----- 83 | 212 | 212 | 0 | 0 |
| 64 ----- 75 | 215 | 0 | 0 | 0 |
| 65 ----- 83 | 233 | 233 | 0 | 0 |
| 66 ----- 83 | 233 | 233 | 0 | 0 |
| 67 ----- 76 | 194 | 0 | 0 | 0 |
| 68 ----- 83 | 166 | 166 | 0 | 0 |
| 70 ----- 78 | 18 | 0 | 0 | 0 |
| 71 ----- 83 | 44 | 44 | 0 | 0 |
| 72 ----- 83 | 226 | 226 | 0 | 0 |
| 73 ----- 79 | 190 | 0 | 0 | 0 |
| 74 ----- 80 | 0 | 0 | 0 | 0 |
| 75 ----- 83 | 215 | 215 | 0 | 0 |
| 76 ----- 83 | 194 | 194 | 0 | 0 |
| 77 ----- 83 | 229 | 229 | 0 | -13 |
| 78 ----- 83 | 18 | 18 | 0 | 0 |
| 79 ----- 81 | 190 | 0 | 0 | 0 |
| 80 ----- 83 | 0 | 0 | 0 | 0 |
| 81 ----- 82 | 190 | 0 | 0 | 0 |
| 83 -----118 | 0 | 0 | 0 | 0 |
| 84 ----- 17 | 238 | 0 | 0 | 0 |

IN STMT 10 PROGRAM RETURNS FROM MAIN PROCEDURE.

```
IN STMT   10  SCALARS AND BLOCK-TRACE:

***** MAIN PROCEDURE DFSCRIP

L=        0          Z=        0          Y=        4          X=       117          P=        ?
M=        1          K=        0          J=       17          I=       119          C=      183
B=      364          A=      366          TOTAL_EVENTS=      118


NON-O PROCEDURE EXECUTION COUNTS:
NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT
CP_ROUTE     0210 00011  CRITICALPAT  0171 00368  BUILD_TREE   0134 00368  TRACE        0120 00366  SEARCH       0014 00001
DFSCRIP      0001 00001

LABEL EXECUTION COUNTS:
NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT
NEXT         0115 00183  OUT          0066 00117  JUMP         0057 00183  LOOP         0037 00183


       COMPILATION STATISTICS  (0244 STATEMENTS)       |                EXECUTION STATISTICS
SECONDS  ERRORS  WARNINGS  PAGES  LINES  CARDS  INCL'S | SECONDS  ERRORS  WARNINGS  PAGES  LINES  CARDS  INCL'S  AUX I/O
    .16       0         1      0      4    468       0 |     .38       5         0      7    285    122       0        0
-------------------------------------------------------+-----------------------------------------------------------------
BYTES   SYMBOL TABLE   INTERMEDIATE CODE    OBJECT CODE |   STATIC CORE   AUTOMATIC CORE    DYNAMIC CORE   TOTAL STORAGE
USED     8730(   9K)       6580(   7K)     12528(  13K) |     346(   1K)   101590( 100K)        0(   0K)  122398( 120K)
UNUSED  53878(  52K)      55699(  54K)    104988( 102K) |    3046(   2K)     3046(   2K)     3046(   2K)    3046(   2K)

THIS PROGRAM MAY BE RERUN WITHOUT CHANGE IN A REGION   2K BYTES SMALLER USING TABLESIZE=  2183
```

APPENDIX 504

Phillips Project, Conventional Method

CRITICAL PATH LENGTH = 256

CRITICAL EVENTS 1 2 4 22 43 61 62
74 80 83 118

| EVENT | EVENT FLOAT |
|---|---|
| 1 | 0 |
| 2 | 0 |
| 3 | 228 |
| 4 | 0 |
| 5 | 240 |
| 6 | 241 |
| 7 | 212 |
| 8 | 231 |
| 9 | 187 |
| 10 | 166 |
| 11 | 188 |
| 12 | 244 |
| 13 | 18 |
| 14 | 44 |
| 15 | 196 |
| 16 | 238 |
| 17 | 238 |
| 18 | 226 |
| 19 | 190 |
| 20 | 240 |
| 21 | 228 |
| 22 | 0 |
| 23 | 240 |
| 24 | 241 |
| 25 | 212 |
| 26 | 232 |
| 27 | 215 |
| 28 | 231 |
| 29 | 225 |
| 30 | 233 |
| 31 | 231 |
| 32 | 233 |
| 33 | 187 |
| 34 | 194 |
| 35 | 166 |
| 36 | 188 |
| 37 | 18 |
| 38 | 44 |
| 39 | 196 |
| 40 | 238 |
| 41 | 226 |
| 42 | 190 |
| 43 | 0 |

| | |
|---|---|
| 44 | 241 |
| 45 | 212 |
| 46 | 232 |
| 47 | 215 |
| 48 | 231 |
| 49 | 225 |
| 50 | 233 |
| 51 | 233 |
| 52 | 187 |
| 53 | 194 |
| 54 | 166 |
| 55 | 188 |
| 56 | 18 |
| 57 | 44 |
| 58 | 226 |
| 59 | 190 |
| 60 | 192 |
| 61 | 0 |
| 62 | 0 |
| 63 | 212 |
| 64 | 215 |
| 65 | 233 |
| 66 | 233 |
| 67 | 194 |
| 68 | 166 |
| 69 | 188 |
| 70 | 18 |
| 71 | 44 |
| 72 | 226 |
| 73 | 190 |
| 74 | 0 |
| 75 | 215 |
| 76 | 194 |
| 77 | 243 |
| 78 | 18 |
| 79 | 190 |
| 80 | 0 |
| 81 | 190 |
| 82 | 190 |
| 83 | 0 |
| 84 | 238 |
| 85 | 190 |
| 86 | 246 |
| 87 | 247 |
| 88 | 212 |
| 89 | 240 |
| 90 | 246 |
| 91 | 245 |
| 92 | 242 |
| 93 | 224 |
| 94 | 247 |
| 95 | 246 |
| 96 | 244 |
| 97 | 246 |
| 98 | 212 |
| 99 | 244 |
| 100 | 247 |
| 101 | 244 |
| 102 | 246 |
| 103 | 240 |

| | |
|---|---|
| 104 | 244 |
| 105 | 246 |
| 106 | 247 |
| 107 | 242 |
| 108 | 239 |
| 109 | 243 |
| 110 | 231 |
| 111 | 237 |
| 112 | 246 |
| 113 | 244 |
| 114 | 246 |
| 115 | 245 |
| 116 | 240 |
| 117 | 247 |
| 118 | 0 |

| ACTIVITY | TOTAL FLOAT | FREE FLOAT | INDEP. FLOAT | SAFETY FLOAT |
|---|---|---|---|---|
| 1 ----- 2 | 0 | 0 | 0 | 0 |
| 2 ----- 83 | 188 | 188 | 188 | 188 |
| 2 ----- 85 | 246 | 56 | 56 | 246 |
| 2 ----- 4 | 0 | 0 | 0 | 0 |
| 2 ----- 86 | 246 | 0 | 0 | 246 |
| 2 ----- 87 | 247 | 0 | 0 | 247 |
| 2 ----- 88 | 212 | 0 | 0 | 212 |
| 2 ----- 89 | 240 | 0 | 0 | 240 |
| 2 ----- 3 | 228 | 0 | 0 | 228 |
| 2 ----- 90 | 246 | 0 | 0 | 246 |
| 2 ----- 5 | 240 | 0 | 0 | 240 |
| 2 ----- 8 | 231 | 0 | 0 | 231 |
| 2 ----- 91 | 245 | 0 | 0 | 245 |
| 2 ----- 6 | 241 | 0 | 0 | 241 |
| 2 ----- 92 | 242 | 0 | 0 | 242 |
| 2 ----- 93 | 224 | 0 | 0 | 224 |
| 2 ----- 94 | 247 | 0 | 0 | 247 |
| 2 ----- 95 | 246 | 0 | 0 | 246 |
| 2 ----- 96 | 244 | 0 | 0 | 244 |
| 2 ----- 97 | 246 | 0 | 0 | 246 |
| 2 ----- 7 | 212 | 0 | 0 | 212 |
| 2 ----- 98 | 212 | 0 | 0 | 212 |
| 2 ----- 99 | 244 | 0 | 0 | 244 |
| 2 -----100 | 247 | 0 | 0 | 247 |
| 2 ----- 9 | 187 | 0 | 0 | 187 |
| 2 ----- 10 | 166 | 0 | 0 | 166 |
| 2 -----101 | 244 | 0 | 0 | 244 |
| 2 -----102 | 246 | 0 | 0 | 246 |
| 2 ----- 11 | 188 | 0 | 0 | 188 |
| 2 ----- 12 | 244 | 0 | 0 | 244 |
| 2 ----- 13 | 18 | 0 | 0 | 18 |
| 2 -----103 | 240 | 0 | 0 | 240 |
| 2 -----104 | 244 | 0 | 0 | 244 |
| 2 -----105 | 246 | 0 | 0 | 246 |
| 2 ----- 14 | 44 | 0 | 0 | 44 |
| 2 ----- 15 | 196 | 0 | 0 | 196 |
| 2 ----- 16 | 238 | 0 | 0 | 238 |
| 2 ----- 17 | 238 | 0 | 0 | 238 |
| 2 ----- 18 | 226 | 0 | 0 | 226 |
| 2 ----- 19 | 190 | 0 | 0 | 190 |
| 2 ----- 20 | 240 | 0 | 0 | 240 |
| 3 ----- 21 | 228 | 0 | 0 | 0 |
| 4 ----- 22 | 0 | 0 | 0 | 0 |
| 5 ----- 23 | 240 | 0 | 0 | 0 |
| 6 ----- 24 | 241 | 0 | 0 | 0 |
| 7 ----- 25 | 212 | 0 | 0 | 0 |
| 8 -----107 | 242 | 0 | 0 | 11 |
| 8 -----108 | 239 | 0 | 0 | 8 |
| 8 ----- 26 | 232 | 0 | 0 | 1 |
| 8 -----109 | 243 | 0 | 0 | 12 |
| 8 -----110 | 231 | 0 | 0 | 0 |
| 8 -----111 | 237 | 0 | 0 | 6 |
| 8 ----- 83 | 241 | 241 | 10 | 10 |
| 9 ----- 83 | 244 | 244 | 57 | 57 |

| | | | | | |
|---|---|---|---|---|---|
| 9 | -----112 | 246 | 0 | 0 | 59 |
| 9 | ----- 27 | 215 | 0 | 0 | 28 |
| 9 | ----- 28 | 231 | 0 | 0 | 44 |
| 9 | -----113 | 244 | 0 | 0 | 57 |
| 9 | -----114 | 246 | 0 | 0 | 59 |
| 9 | ----- 29 | 225 | 0 | 0 | 38 |
| 9 | ----- 30 | 233 | 0 | 0 | 46 |
| 9 | ----- 31 | 231 | 0 | 0 | 44 |
| 9 | ----- 32 | 233 | 0 | 0 | 46 |
| 9 | -----115 | 245 | 0 | 0 | 58 |
| 9 | ----- 33 | 187 | 0 | 0 | 0 |
| 9 | -----116 | 240 | 0 | 0 | 53 |
| 9 | ----- 34 | 194 | 0 | 0 | 7 |
| 10 | ----- 35 | 166 | 0 | 0 | 0 |
| 11 | ----- 36 | 188 | 0 | 0 | 0 |
| 12 | ----- 83 | 244 | 244 | 0 | 0 |
| 13 | ----- 37 | 18 | 0 | 0 | 0 |
| 14 | ----- 38 | 44 | 0 | 0 | 0 |
| 15 | ----- 39 | 196 | 0 | 0 | 0 |
| 16 | ----- 40 | 238 | 0 | 0 | 0 |
| 17 | ----- 83 | 238 | 238 | 0 | 0 |
| 18 | ----- 41 | 226 | 0 | 0 | 0 |
| 19 | ----- 42 | 190 | 0 | 0 | 0 |
| 20 | ----- 83 | 240 | 240 | 0 | 0 |
| 21 | ----- 83 | 228 | 228 | 0 | 0 |
| 22 | ----- 43 | 0 | 0 | 0 | 0 |
| 23 | ----- 83 | 240 | 240 | 0 | 0 |
| 24 | ----- 44 | 241 | 0 | 0 | 0 |
| 25 | ----- 45 | 212 | 0 | 0 | 0 |
| 26 | ----- 46 | 232 | 0 | 0 | 0 |
| 27 | ----- 47 | 215 | 0 | 0 | 0 |
| 28 | ----- 48 | 231 | 0 | 0 | 0 |
| 29 | ----- 49 | 225 | 0 | 0 | 0 |
| 30 | ----- 50 | 233 | 0 | 0 | 0 |
| 31 | ----- 83 | 231 | 231 | 0 | 0 |
| 32 | ----- 51 | 233 | 0 | 0 | 0 |
| 33 | ----- 52 | 187 | 0 | 0 | 0 |
| 34 | ----- 53 | 194 | 0 | 0 | 0 |
| 35 | ----- 54 | 166 | 0 | 0 | 0 |
| 36 | ----- 55 | 188 | 0 | 0 | 0 |
| 37 | ----- 56 | 18 | 0 | 0 | 0 |
| 38 | ----- 57 | 44 | 0 | 0 | 0 |
| 39 | ----- 83 | 196 | 196 | 0 | 0 |
| 40 | ----- 83 | 238 | 238 | 0 | 0 |
| 41 | ----- 58 | 226 | 0 | 0 | 0 |
| 42 | ----- 59 | 190 | 0 | 0 | 0 |
| 43 | ----- 83 | 212 | 212 | 212 | 212 |
| 43 | ----- 60 | 192 | 0 | 0 | 192 |
| 43 | ----- 61 | 0 | 0 | 0 | 0 |
| 44 | ----- 83 | 241 | 241 | 0 | 0 |
| 45 | ----- 63 | 212 | 0 | 0 | 0 |
| 46 | ----- 83 | 232 | 232 | 0 | 0 |
| 47 | ----- 64 | 215 | 0 | 0 | 0 |
| 48 | ----- 83 | 231 | 231 | 0 | 0 |
| 49 | ----- 83 | 225 | 225 | 0 | 0 |
| 50 | ----- 65 | 233 | 0 | 0 | 0 |
| 51 | ----- 66 | 233 | 0 | 0 | 0 |
| 52 | ----- 83 | 187 | 187 | 0 | 0 |
| 53 | ----- 67 | 194 | 0 | 0 | 0 |
| 54 | ----- 68 | 166 | 0 | 0 | 0 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 55 | ----- | 69 | 188 | 0 | 0 | 0 |
| 56 | ----- | 70 | 18 | 0 | 0 | 0 |
| 57 | ----- | 71 | 44 | 0 | 0 | 0 |
| 58 | ----- | 72 | 226 | 0 | 0 | 0 |
| 59 | ----- | 73 | 190 | 0 | 0 | 0 |
| 60 | ----- | 83 | 192 | 192 | 0 | 0 |
| 61 | ----- | 74 | 36 | 36 | 36 | 36 |
| 61 | ----- | 62 | 0 | 0 | 0 | 0 |
| 62 | ----- | 83 | 96 | 96 | 96 | 96 |
| 62 | ----- | 74 | 0 | 0 | 0 | 0 |
| 63 | ----- | 83 | 212 | 212 | 0 | 0 |
| 64 | ----- | 75 | 215 | 0 | 0 | 0 |
| 65 | ----- | 83 | 233 | 233 | 0 | 0 |
| 66 | ----- | 83 | 233 | 233 | 0 | 0 |
| 67 | ----- | 76 | 194 | 0 | 0 | 0 |
| 68 | ----- | 83 | 166 | 166 | 0 | 0 |
| 70 | ----- | 78 | 18 | 0 | 0 | 0 |
| 71 | ----- | 83 | 44 | 44 | 0 | 0 |
| 72 | ----- | 83 | 226 | 226 | 0 | 0 |
| 73 | ----- | 79 | 190 | 0 | 0 | 0 |
| 74 | ----- | 80 | 0 | 0 | 0 | 0 |
| 75 | ----- | 83 | 215 | 215 | 0 | 0 |
| 76 | ----- | 83 | 194 | 194 | 0 | 0 |
| 77 | ----- | 83 | 243 | 243 | 0 | 0 |
| 78 | ----- | 83 | 18 | 18 | 0 | 0 |
| 79 | ----- | 81 | 190 | 0 | 0 | 0 |
| 80 | ----- | 83 | 0 | 0 | 0 | 0 |
| 81 | ----- | 82 | 190 | 0 | 0 | 0 |
| 83 | ----- | 118 | 0 | 0 | 0 | 0 |
| 84 | ----- | 17 | 238 | 0 | 0 | 0 |

IN STMT 10 PROGRAM RETURNS FROM MAIN PROCEDURE.

```
IN STMT    10  SCALARS AND BLOCK-TRACE:

***** MAIN PROCEDURE CONCRIP

L=        ?              Z=       0            Y=       4             X=      117           P=        ?
M=        1              K=       0            J=      17             I=      119           C=        ?
B=      364              A=     366            TOTAL_EVENTS=      118


NON-0 PROCEDURE EXECUTION COUNTS:
NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT
TEST         0189 00278  CALCULATE    0169 00449  PASS         0120 00002  TRACE        0107 00366  EXECUTE      0014 00001
CONCRIP      0001 00001

LABEL EXECUTION COUNTS:
NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT
AHEAD        0201 00278  CONTINUE     0186 00449  FURTHER      0182 00046  HOP          0166 00001  SKIP         0149 00001
BACK         0132 00004  JUMP         0105 00001  NEXT         0101 00183  OUT          0061 00117  LOOP         0033 00183


       COMPILATION STATISTICS  (0216 STATEMENTS)     |                    EXECUTION STATISTICS
SECONDS  ERRORS  WARNINGS  PAGES  LINES  CARDS  INCL'S | SECONDS  ERRORS  WARNINGS  PAGES  LINES  CARDS  INCL'S  AUX I/O
    .12       0         1      0      4    401       0 |     .33       0         0      7    290    122       0        0
-------------------------------------------------------+-------------------------------------------------------------------
BYTES   SYMBOL TABLE    INTERMEDIATE CODE    OBJECT CODE |    STATIC CORE   AUTOMATIC CORE    DYNAMIC CORE   TOTAL STORAGE
USED     8199(   9K)        5638(   6K)     10656(  11K) |     342(    1K)   20934(  21K)        0(    0K)   39350(  39K)
UNUSED  11401(  11K)       13633(  13K)     21362(  20K) |      78(    0K)      78(   0K)       78(    0K)      78(   0K)

THIS PROGRAM MAY BE RERUN WITHOUT CHANGE IN A REGION   0K BYTES SMALLER USING TABLESIZE=  2050
```

APPENDIX 505

Motel Construction Project, Conventional Method

PROJECT

CRITICAL PATH LENGTH = 263

CRITICAL EVENTS

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 8 | 18 | 20 | 26 |
| 30 | 33 | 34 | 35 | 37 | 38 | 39 |
| 44 | 47 | 48 | 53 | 61 | 62 | 71 |
| 73 | 78 | 79 | 80 | 81 | 83 | 84 |
| 85 | 87 | 89 | 91 | 97 | 99 | 103 |
| 105 | 106 | 122 | 129 | 134 | 137 | 138 |
| 139 | 253 | 254 | | | | |

| EVENT | EVENT FLOAT |
|---|---|
| 1 | 0 |
| 2 | 0 |
| 3 | 0 |
| 4 | 23 |
| 5 | 50 |
| 6 | 11 |
| 7 | 4 |
| 8 | 0 |
| 9 | 42 |
| 10 | 11 |
| 11 | 23 |
| 12 | 23 |
| 13 | 2 |
| 14 | 11 |
| 15 | 11 |
| 16 | 2 |
| 17 | 12 |
| 18 | 0 |
| 19 | 13 |
| 20 | 0 |
| 21 | 119 |
| 22 | 131 |
| 23 | 50 |
| 24 | 11 |
| 25 | 4 |
| 26 | 0 |
| 27 | 42 |
| 28 | 11 |
| 29 | 4 |
| 30 | 0 |
| 31 | 42 |
| 32 | 4 |
| 33 | 0 |
| 34 | 0 |
| 35 | 0 |
| 36 | 42 |
| 37 | 0 |
| 38 | 0 |

| | |
|---|---|
| 39 | 0 |
| 40 | 108 |
| 41 | 108 |
| 42 | 131 |
| 43 | 119 |
| 44 | 0 |
| 45 | 41 |
| 46 | 41 |
| 47 | 0 |
| 48 | 0 |
| 49 | 11 |
| 50 | 11 |
| 51 | 42 |
| 52 | 41 |
| 53 | 0 |
| 54 | 11 |
| 55 | 41 |
| 56 | 44 |
| 57 | 12 |
| 58 | 11 |
| 59 | 12 |
| 60 | 13 |
| 61 | 0 |
| 62 | 0 |
| 63 | 44 |
| 64 | 47 |
| 65 | 41 |
| 66 | 44 |
| 67 | 44 |
| 68 | 47 |
| 69 | 47 |
| 70 | 50 |
| 71 | 0 |
| 72 | 13 |
| 73 | 0 |
| 74 | 11 |
| 75 | 47 |
| 76 | 50 |
| 77 | 15 |
| 78 | 0 |
| 79 | 0 |
| 80 | 0 |
| 81 | 0 |
| 82 | 76 |
| 83 | 0 |
| 84 | 0 |
| 85 | 0 |
| 86 | 22 |
| 87 | 0 |
| 88 | 22 |
| 89 | 0 |
| 90 | 15 |
| 91 | 0 |
| 92 | 24 |
| 93 | 27 |
| 94 | 15 |
| 95 | 15 |
| 96 | 15 |
| 97 | 0 |
| 98 | 10 |

| | |
|---|---|
| 99 | 0 |
| 100 | 15 |
| 101 | 10 |
| 102 | 10 |
| 103 | 0 |
| 104 | 5 |
| 105 | 0 |
| 106 | 0 |
| 107 | 5 |
| 108 | 5 |
| 109 | 10 |
| 110 | 10 |
| 111 | 15 |
| 112 | 15 |
| 113 | 27 |
| 114 | 48 |
| 115 | 48 |
| 116 | 15 |
| 117 | 15 |
| 118 | 10 |
| 119 | 10 |
| 120 | 5 |
| 121 | 5 |
| 122 | 0 |
| 123 | 15 |
| 124 | 31 |
| 125 | 10 |
| 126 | 10 |
| 127 | 5 |
| 128 | 5 |
| 129 | 0 |
| 130 | 10 |
| 131 | 21 |
| 132 | 5 |
| 133 | 5 |
| 134 | 0 |
| 135 | 5 |
| 136 | 11 |
| 137 | 0 |
| 138 | 0 |
| 139 | 0 |
| 140 | 60 |
| 141 | 169 |
| 142 | 169 |
| 143 | 169 |
| 144 | 77 |
| 145 | 77 |
| 146 | 77 |
| 147 | 77 |
| 148 | 77 |
| 149 | 77 |
| 150 | 60 |
| 151 | 60 |
| 152 | 80 |
| 153 | 60 |
| 154 | 82 |
| 155 | 82 |
| 156 | 82 |
| 157 | 82 |
| 158 | 82 |

| | |
|---|---|
| 159 | 82 |
| 160 | 82 |
| 161 | 82 |
| 162 | 82 |
| 163 | 82 |
| 164 | 76 |
| 165 | 108 |
| 166 | 108 |
| 167 | 108 |
| 168 | 76 |
| 169 | 76 |
| 170 | 80 |
| 171 | 76 |
| 172 | 64 |
| 173 | 108 |
| 174 | 116 |
| 175 | 108 |
| 176 | 104 |
| 177 | 92 |
| 178 | 92 |
| 179 | 106 |
| 180 | 104 |
| 181 | 92 |
| 182 | 92 |
| 183 | 104 |
| 184 | 92 |
| 185 | 97 |
| 186 | 92 |
| 187 | 104 |
| 188 | 92 |
| 189 | 63 |
| 190 | 107 |
| 191 | 100 |
| 192 | 104 |
| 193 | 104 |
| 194 | 95 |
| 195 | 92 |
| 196 | 63 |
| 197 | 63 |
| 198 | 65 |
| 199 | 63 |
| 200 | 63 |
| 201 | 64 |
| 202 | 95 |
| 203 | 104 |
| 204 | 95 |
| 205 | 64 |
| 206 | 64 |
| 207 | 79 |
| 208 | 63 |
| 209 | 92 |
| 210 | 97 |
| 211 | 92 |
| 212 | 92 |
| 213 | 102 |
| 214 | 94 |
| 215 | 92 |
| 216 | 92 |
| 217 | 99 |
| 218 | 92 |

| | |
|---|---|
| 219 | 92 |
| 220 | 63 |
| 221 | 71 |
| 222 | 68 |
| 223 | 63 |
| 224 | 63 |
| 225 | 63 |
| 226 | 63 |
| 227 | 63 |
| 228 | 91 |
| 229 | 79 |
| 230 | 64 |
| 231 | 64 |
| 232 | 79 |
| 233 | 91 |
| 234 | 91 |
| 235 | 93 |
| 236 | 92 |
| 237 | 91 |
| 238 | 79 |
| 239 | 64 |
| 240 | 64 |
| 241 | 79 |
| 242 | 79 |
| 243 | 90 |
| 244 | 90 |
| 245 | 79 |
| 246 | 64 |
| 247 | 79 |
| 248 | 64 |
| 249 | 11 |
| 250 | 79 |
| 251 | 91 |
| 252 | 91 |
| 253 | 0 |
| 254 | 0 |

| ACTIVITY | | TOTAL FLOAT | FREE FLOAT | INDEP FLOAT | SAFETY FLOAT |
|---|---|---|---|---|---|
| 2 | ----- 3 | 0 | 0 | 0 | 0 |
| 2 | ----- 4 | 23 | 0 | 0 | 23 |
| 3 | ----- 5 | 50 | 0 | 0 | 50 |
| 3 | ----- 6 | 11 | 0 | 0 | 11 |
| 3 | ----- 7 | 4 | 0 | 0 | 4 |
| 3 | ----- 8 | 0 | 0 | 0 | 0 |
| 3 | ----- 9 | 42 | 0 | 0 | 42 |
| 3 | ----- 10 | 11 | 0 | 0 | 11 |
| 4 | ----- 11 | 23 | 0 | 0 | 0 |
| 4 | -----141 | 169 | 0 | 0 | 146 |
| 5 | ----- 23 | 50 | 0 | 0 | 0 |
| 6 | ----- 24 | 11 | 0 | 0 | 0 |
| 7 | ----- 25 | 4 | 0 | 0 | 0 |
| 8 | ----- 26 | 0 | 0 | 0 | 0 |
| 9 | ----- 27 | 42 | 0 | 0 | 0 |
| 10 | ----- 28 | 11 | 0 | 0 | 0 |
| 11 | ----- 12 | 23 | 0 | 0 | 0 |
| 13 | ----- 16 | 2 | 0 | 0 | 0 |
| 14 | ----- 15 | 11 | 0 | 0 | 0 |
| 15 | ----- 17 | 12 | 0 | 0 | 1 |
| 16 | ----- 18 | 2 | 2 | 0 | 0 |
| 17 | ----- 19 | 13 | 0 | 0 | 1 |
| 18 | ----- 20 | 0 | 0 | 0 | 0 |
| 21 | ----- 43 | 119 | 0 | 0 | 0 |
| 22 | ----- 42 | 131 | 0 | 0 | 0 |
| 24 | ----- 37 | 11 | 11 | 0 | 0 |
| 25 | ----- 29 | 4 | 0 | 0 | 0 |
| 26 | ----- 30 | 0 | 0 | 0 | 0 |
| 27 | ----- 31 | 42 | 0 | 0 | 0 |
| 29 | ----- 32 | 4 | 0 | 0 | 0 |
| 30 | ----- 33 | 0 | 0 | 0 | 0 |
| 31 | ----- 36 | 42 | 0 | 0 | 0 |
| 33 | ----- 34 | 0 | 0 | 0 | 0 |
| 34 | ----- 35 | 0 | 0 | 0 | 0 |
| 37 | ----- 38 | 0 | 0 | 0 | 0 |
| 38 | ----- 39 | 0 | 0 | 0 | 0 |
| 39 | ----- 44 | 0 | 0 | 0 | 0 |
| 39 | ----- 40 | 108 | 0 | 0 | 108 |
| 39 | -----177 | 92 | 0 | 0 | 92 |
| 41 | -----165 | 108 | 0 | 0 | 0 |
| 42 | ----- 43 | 132 | 13 | 0 | 1 |
| 43 | ----- 41 | 119 | 11 | 0 | 0 |
| 44 | ----- 45 | 41 | 0 | 0 | 41 |
| 44 | ----- 47 | 0 | 0 | 0 | 0 |
| 45 | ----- 51 | 42 | 0 | 0 | 1 |
| 46 | ----- 52 | 41 | 0 | 0 | 0 |
| 48 | ----- 53 | 0 | 0 | 0 | 0 |
| 50 | ----- 54 | 11 | 0 | 0 | 0 |
| 53 | ----- 61 | 0 | 0 | 0 | 0 |
| 54 | ----- 58 | 11 | 0 | 0 | 0 |
| 55 | ----- 65 | 41 | 0 | 0 | 0 |
| 56 | ----- 63 | 44 | 0 | 0 | 0 |
| 57 | ----- 59 | 12 | 0 | 0 | 0 |
| 58 | ----- 74 | 11 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 60 ----- 72 | 13 | 0 | 0 | 0 |
| 62 ----- 71 | 0 | 0 | 0 | 0 |
| 64 ----- 69 | 47 | 0 | 0 | 0 |
| 66 ----- 67 | 44 | 0 | 0 | 0 |
| 68 ----- 75 | 47 | 0 | 0 | 0 |
| 70 ----- 76 | 50 | 0 | 0 | 0 |
| 71 ----- 77 | 15 | 0 | 0 | 15 |
| 73 ----- 78 | 0 | 0 | 0 | 0 |
| 74 ----- 79 | 11 | 11 | 0 | 0 |
| 76 -----105 | 50 | 50 | 0 | 0 |
| 78 ----- 81 | 0 | 0 | 0 | 0 |
| 79 ----- 80 | 0 | 0 | 0 | 0 |
| 80 ----- 83 | 0 | 0 | 0 | 0 |
| 81 ----- 82 | 76 | 0 | 0 | 76 |
| 82 -----164 | 76 | 0 | 0 | 0 |
| 84 ----- 85 | 0 | 0 | 0 | 0 |
| 86 ----- 88 | 22 | 0 | 0 | 0 |
| 87 ----- 89 | 0 | 0 | 0 | 0 |
| 88 ----- 92 | 24 | 0 | 0 | 2 |
| 90 ----- 94 | 15 | 0 | 0 | 0 |
| 91 ----- 97 | 0 | 0 | 0 | 0 |
| 92 ----- 93 | 27 | 0 | 0 | 3 |
| 93 -----113 | 27 | 0 | 0 | 0 |
| 95 ----- 96 | 15 | 0 | 0 | 0 |
| 98 -----101 | 10 | 0 | 0 | 0 |
| 99 -----103 | 0 | 0 | 0 | 0 |
| 100 -----111 | 15 | 0 | 0 | 0 |
| 102 -----109 | 10 | 0 | 0 | 0 |
| 104 -----107 | 5 | 0 | 0 | 0 |
| 105 -----106 | 0 | 0 | 0 | 0 |
| 106 -----122 | 0 | 0 | 0 | 0 |
| 108 -----120 | 5 | 0 | 0 | 0 |
| 110 -----118 | 10 | 0 | 0 | 0 |
| 112 -----116 | 15 | 0 | 0 | 0 |
| 114 -----115 | 48 | 0 | 0 | 0 |
| 115 -----124 | 48 | 17 | 0 | 0 |
| 117 -----123 | 15 | 0 | 0 | 0 |
| 119 -----125 | 10 | 0 | 0 | 0 |
| 121 -----127 | 5 | 0 | 0 | 0 |
| 122 -----129 | 0 | 0 | 0 | 0 |
| 124 -----131 | 31 | 10 | 0 | 0 |
| 126 -----130 | 10 | 0 | 0 | 0 |
| 128 -----132 | 5 | 0 | 0 | 0 |
| 129 -----134 | 0 | 0 | 0 | 0 |
| 131 -----136 | 21 | 10 | 0 | 0 |
| 133 -----135 | 5 | 0 | 0 | 0 |
| 134 -----137 | 0 | 0 | 0 | 0 |
| 136 -----138 | 11 | 11 | 0 | 0 |
| 137 -----138 | 0 | 0 | 0 | 0 |
| 138 -----253 | 0 | 0 | 0 | 0 |
| 140 -----150 | 60 | 0 | 0 | 0 |
| 141 -----142 | 169 | 0 | 0 | 0 |
| 142 -----143 | 169 | 0 | 0 | 0 |
| 144 -----145 | 77 | 0 | 0 | 0 |
| 145 -----146 | 77 | 0 | 0 | 0 |
| 146 -----147 | 77 | 0 | 0 | 0 |
| 147 -----148 | 77 | 0 | 0 | 0 |
| 148 -----149 | 77 | 0 | 0 | 0 |
| 149 -----253 | 77 | 77 | 0 | 0 |
| 151 -----153 | 60 | 0 | 0 | 0 |

| | | | | | |
|---|---|---|---|---|---|
| 152 | -----153 | 80 | 20 | 0 | 0 |
| 153 | -----253 | 60 | 60 | 0 | 0 |
| 154 | -----155 | 82 | 0 | 0 | 0 |
| 155 | -----156 | 82 | 0 | 0 | 0 |
| 156 | -----157 | 82 | 0 | 0 | 0 |
| 157 | -----158 | 82 | 0 | 0 | 0 |
| 158 | -----159 | 82 | 0 | 0 | 0 |
| 159 | -----160 | 82 | 0 | 0 | 0 |
| 160 | -----161 | 82 | 0 | 0 | 0 |
| 161 | -----163 | 82 | 0 | 0 | 0 |
| 162 | -----161 | 82 | 0 | 0 | 0 |
| 163 | -----253 | 82 | 82 | 0 | 0 |
| 164 | -----168 | 76 | 0 | 0 | 0 |
| 166 | -----167 | 108 | 0 | 0 | 0 |
| 167 | -----174 | 122 | 6 | 0 | 14 |
| 167 | -----175 | 108 | 0 | 0 | 0 |
| 168 | -----170 | 80 | 0 | 0 | 4 |
| 168 | -----169 | 76 | 0 | 0 | 0 |
| 168 | -----171 | 76 | 0 | 0 | 0 |
| 172 | -----231 | 64 | 0 | 0 | 0 |
| 174 | -----205 | 116 | 52 | 0 | 0 |
| 175 | -----173 | 108 | 0 | 0 | 0 |
| 176 | -----193 | 104 | 0 | 0 | 0 |
| 177 | -----178 | 92 | 0 | 0 | 0 |
| 178 | -----179 | 106 | 0 | 0 | 14 |
| 178 | -----180 | 104 | 0 | 0 | 12 |
| 178 | -----181 | 92 | 0 | 0 | 0 |
| 178 | -----182 | 92 | 0 | 0 | 0 |
| 181 | -----185 | 97 | 0 | 0 | 5 |
| 182 | -----192 | 104 | 0 | 0 | 12 |
| 183 | -----187 | 104 | 0 | 0 | 0 |
| 184 | -----186 | 92 | 0 | 0 | 0 |
| 188 | -----195 | 92 | 0 | 0 | 0 |
| 189 | -----197 | 63 | 0 | 0 | 0 |
| 190 | -----201 | 107 | 43 | 0 | 0 |
| 191 | -----194 | 100 | 5 | 0 | 0 |
| 194 | -----202 | 95 | 0 | 0 | 0 |
| 196 | -----199 | 63 | 0 | 0 | 0 |
| 197 | -----198 | 65 | 0 | 0 | 2 |
| 198 | -----209 | 92 | 0 | 0 | 27 |
| 200 | -----208 | 63 | 0 | 0 | 0 |
| 201 | -----206 | 64 | 0 | 0 | 0 |
| 203 | -----204 | 104 | 9 | 0 | 0 |
| 204 | -----205 | 95 | 31 | 0 | 0 |
| 205 | -----230 | 64 | 0 | 0 | 0 |
| 207 | -----250 | 79 | 0 | 0 | 0 |
| 208 | -----220 | 63 | 0 | 0 | 0 |
| 209 | -----210 | 97 | 0 | 0 | 5 |
| 209 | -----211 | 92 | 0 | 0 | 0 |
| 209 | -----213 | 102 | 0 | 0 | 10 |
| 210 | -----217 | 99 | 0 | 0 | 2 |
| 211 | -----215 | 92 | 0 | 0 | 0 |
| 211 | -----214 | 94 | 0 | 0 | 2 |
| 212 | -----216 | 92 | 0 | 0 | 0 |
| 218 | -----219 | 92 | 0 | 0 | 0 |
| 219 | -----253 | 92 | 92 | 0 | 0 |
| 220 | -----221 | 71 | 0 | 0 | 8 |
| 220 | -----222 | 68 | 0 | 0 | 5 |
| 220 | -----223 | 63 | 0 | 0 | 0 |
| 224 | -----225 | 63 | 0 | 0 | 0 |

| | | | | | |
|---|---|---|---|---|---|
| 225 | -----226 | 63 | 0 | 0 | 0 |
| 226 | -----227 | 63 | 0 | 0 | 0 |
| 227 | -----253 | 63 | 63 | 0 | 0 |
| 228 | -----233 | 91 | 0 | 0 | 0 |
| 229 | -----232 | 79 | 0 | 0 | 0 |
| 231 | -----239 | 64 | 0 | 0 | 0 |
| 232 | -----238 | 79 | 0 | 0 | 0 |
| 233 | -----234 | 91 | 0 | 0 | 0 |
| 233 | -----235 | 93 | 0 | 0 | 2 |
| 233 | -----236 | 92 | 0 | 0 | 1 |
| 237 | -----251 | 91 | 0 | 0 | 0 |
| 238 | -----243 | 90 | 0 | 0 | 11 |
| 238 | -----244 | 90 | 0 | 0 | 11 |
| 238 | -----241 | 79 | 0 | 0 | 0 |
| 239 | -----240 | 64 | 0 | 0 | 0 |
| 240 | -----246 | 64 | 0 | 0 | 0 |
| 241 | -----242 | 79 | 0 | 0 | 0 |
| 242 | -----245 | 79 | 0 | 0 | 0 |
| 245 | -----247 | 79 | 0 | 0 | 0 |
| 246 | -----248 | 64 | 0 | 0 | 0 |
| 247 | -----253 | 79 | 79 | 0 | 0 |
| 248 | -----253 | 64 | 64 | 0 | 0 |
| 249 | ----- 49 | 11 | 0 | 0 | 0 |
| 251 | -----252 | 91 | 0 | 0 | 0 |
| 252 | -----253 | 91 | 91 | 0 | 0 |
| 253 | -----254 | 0 | 0 | 0 | 0 |

IN STMT 10 PROGRAM RETURNS FROM MAIN PROCEDURE.

```
IN STMT   10  SCALARS AND BLOCK-TRACE:

***** MAIN PROCEDURE CONCRIP

L=         ?          Z=      10          Y=       3             X=    253           P=     ?
M=         1          K=       0          J=     254             I=    255           C=     ?
B=         0          A=     732          TOTAL_EVENTS=     254


NON-O PROCEDURE EXECUTION COUNTS:
NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT
TEST         0189 01927  CALCULATE    0169 03097  PASS         0120 00002  TRACE        0107 00732  EXECUTE      0014 00001
CONCRIP      0001 00001

LABEL EXECUTION COUNTS:
NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT  NAME         STMT COUNT
AHEAD        0201 01927  CONTINUE     0186 03097  FURTHER      0182 01423  HOP          0166 00001  SKIP         0149 00001
BACK         0132 00016  JUMP         0105 00001  NEXT         0101 00366  OUT          0061 00253  LOOP         0033 00366


       COMPILATION STATISTICS  (0216 STATEMENTS)  |                     EXECUTION STATISTICS
SECONDS  ERRORS  WARNINGS  PAGES  LINES  CARDS  INCL'S | SECONDS  ERRORS  WARNINGS  PAGES  LINES  CARDS  INCL'S  AUX I/O
    .12       0         1      0      4   401.       0 |     .83       0         0     10    487    254       0        0
-------------------------------------------------------+-----------------------------------------------------------------
BYTES   SYMBOL TABLE  INTERMEDIATE CODE   OBJECT CODE  |  STATIC CORE   AUTOMATIC CORE   DYNAMIC CORE   TOTAL STORAGE
USED     8199(  9K)      5638(  6K)      10656( 11K)   |    342(  1K)     39430( 39K)        0(  0K)    57846( 57K)
UNUSED  33929( 33K)     36161( 35K)      66418( 64K)   |  26638( 26K)     26638( 26K)    26638( 26K)    26638( 26K)

THIS PROGRAM MAY BE RERUN WITHOUT CHANGE IN A REGION  26K BYTES SMALLER USING TABLESIZE=  2050
```

APPENDIX 506

Motel Construction Project DFS

CRITICAL PATH LENGTH = 263

CRITICAL PATH

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 8 | 26 | 30 | 33 |
| 34 | 35 | 18 | 20 | 37 | 38 | 39 |
| 44 | 47 | 48 | 53 | 61 | 62 | 71 |
| 73 | 78 | 81 | 80 | 83 | 139 | 84 |
| 85 | 87 | 89 | 91 | 97 | 99 | 103 |
| 105 | 106 | 122 | 129 | 134 | 137 | 138 |
| 253 | 254 | 78 | 79 | 80 | 83 | 139 |
| 84 | 85 | 87 | 89 | 91 | 97 | 99 |
| 103 | 105 | 106 | 122 | 129 | 134 | 137 |
| 138 | 253 | 254 | | | | |

| EVENT | EVENT FLOAT |
|---|---|
| 1 | 0 |
| 2 | 0 |
| 3 | 0 |
| 4 | 23 |
| 5 | 50 |
| 6 | 11 |
| 7 | 4 |
| 8 | 0 |
| 9 | 42 |
| 10 | 11 |
| 11 | 23 |
| 12 | 23 |
| 13 | 2 |
| 14 | 11 |
| 15 | 11 |
| 16 | 2 |
| 17 | 12 |
| 18 | 0 |
| 19 | 13 |
| 20 | 0 |
| 21 | 119 |
| 22 | 131 |
| 23 | 50 |
| 24 | 11 |
| 25 | 4 |
| 26 | 0 |
| 27 | 42 |
| 28 | 11 |
| 29 | 4 |
| 30 | 0 |
| 31 | 42 |
| 32 | 4 |
| 33 | 0 |
| 34 | 0 |
| 35 | 0 |
| 36 | 42 |
| 37 | 0 |
| 38 | 0 |
| 39 | 0 |

| | |
|---|---|
| 40 | 108 |
| 41 | 108 |
| 42 | 131 |
| 43 | 119 |
| 44 | 0 |
| 45 | 41 |
| 46 | 41 |
| 47 | 0 |
| 48 | 0 |
| 49 | 11 |
| 50 | 11 |
| 51 | 42 |
| 52 | 41 |
| 53 | 0 |
| 54 | 11 |
| 55 | 41 |
| 56 | 44 |
| 57 | 12 |
| 58 | 11 |
| 59 | 12 |
| 60 | 13 |
| 61 | 0 |
| 62 | 0 |
| 63 | 44 |
| 64 | 47 |
| 65 | 41 |
| 66 | 44 |
| 67 | 44 |
| 68 | 47 |
| 69 | 47 |
| 70 | 50 |
| 71 | 0 |
| 72 | 13 |
| 73 | 0 |
| 74 | 11 |
| 75 | 47 |
| 76 | 50 |
| 77 | 15 |
| 78 | 0 |
| 79 | 0 |
| 80 | 0 |
| 81 | 0 |
| 82 | 76 |
| 83 | 0 |
| 84 | 0 |
| 85 | 0 |
| 86 | 22 |
| 87 | 0 |
| 88 | 22 |
| 89 | 0 |
| 90 | 15 |
| 91 | 0 |
| 92 | 24 |
| 93 | 27 |
| 94 | 15 |
| 95 | 15 |
| 96 | 15 |
| 97 | 0 |
| 98 | 10 |
| 99 | 0 |

| | |
|---|---|
| 100 | 15 |
| 101 | 10 |
| 102 | 10 |
| 103 | 0 |
| 104 | 5 |
| 105 | 0 |
| 106 | 0 |
| 107 | 5 |
| 108 | 5 |
| 109 | 10 |
| 110 | 10 |
| 111 | 15 |
| 112 | 15 |
| 113 | 27 |
| 114 | 48 |
| 115 | 48 |
| 116 | 15 |
| 117 | 15 |
| 118 | 10 |
| 119 | 10 |
| 120 | 5 |
| 121 | 5 |
| 122 | 0 |
| 123 | 15 |
| 124 | 31 |
| 125 | 10 |
| 126 | 10 |
| 127 | 5 |
| 128 | 5 |
| 129 | 0 |
| 130 | 10 |
| 131 | 21 |
| 132 | 5 |
| 133 | 5 |
| 134 | 0 |
| 135 | 5 |
| 136 | 11 |
| 137 | 0 |
| 138 | 0 |
| 139 | 0 |
| 140 | 60 |
| 141 | 169 |
| 142 | 169 |
| 143 | 169 |
| 144 | 77 |
| 145 | 77 |
| 146 | 77 |
| 147 | 77 |
| 148 | 77 |
| 149 | 77 |
| 150 | 60 |
| 151 | 60 |
| 152 | 80 |
| 153 | 60 |
| 154 | 82 |
| 155 | 82 |
| 156 | 82 |
| 157 | 82 |
| 158 | 82 |
| 159 | 82 |

| | |
|---|---|
| 160 | 82 |
| 161 | 82 |
| 162 | 82 |
| 163 | 82 |
| 164 | 76 |
| 165 | 108 |
| 166 | 108 |
| 167 | 108 |
| 168 | 76 |
| 169 | 76 |
| 170 | 80 |
| 171 | 76 |
| 172 | 64 |
| 173 | 108 |
| 174 | 116 |
| 175 | 108 |
| 176 | 104 |
| 177 | 92 |
| 178 | 92 |
| 179 | 106 |
| 180 | 104 |
| 181 | 92 |
| 182 | 92 |
| 183 | 104 |
| 184 | 92 |
| 185 | 97 |
| 186 | 92 |
| 187 | 104 |
| 188 | 92 |
| 189 | 63 |
| 190 | 107 |
| 191 | 100 |
| 192 | 104 |
| 193 | 104 |
| 194 | 95 |
| 195 | 92 |
| 196 | 63 |
| 197 | 63 |
| 198 | 65 |
| 199 | 63 |
| 200 | 63 |
| 201 | 64 |
| 202 | 95 |
| 203 | 104 |
| 204 | 95 |
| 205 | 64 |
| 206 | 64 |
| 207 | 79 |
| 208 | 63 |
| 209 | 92 |
| 210 | 97 |
| 211 | 92 |
| 212 | 92 |
| 213 | 102 |
| 214 | 94 |
| 215 | 92 |
| 216 | 92 |
| 217 | 99 |
| 218 | 92 |
| 219 | 92 |

| | |
|---|---|
| 220 | 63 |
| 221 | 71 |
| 222 | 68 |
| 223 | 63 |
| 224 | 63 |
| 225 | 63 |
| 226 | 63 |
| 227 | 63 |
| 228 | 91 |
| 229 | 79 |
| 230 | 64 |
| 231 | 64 |
| 232 | 79 |
| 233 | 91 |
| 234 | 91 |
| 235 | 93 |
| 236 | 92 |
| 237 | 91 |
| 238 | 79 |
| 239 | 64 |
| 240 | 64 |
| 241 | 79 |
| 242 | 79 |
| 243 | 90 |
| 244 | 90 |
| 245 | 79 |
| 246 | 64 |
| 247 | 79 |
| 248 | 64 |
| 249 | 11 |
| 250 | 79 |
| 251 | 91 |
| 252 | 91 |
| 253 | 0 |
| 254 | 0 |

| ACTIVITY | TOTAL FLOAT | FREE FLOAT | INDEP. FLOAT | SAFETY FLOAT |
|---|---|---|---|---|
| 2 ----- 3 | 0 | 0 | 0 | 0 |
| 2 ----- 4 | 23 | 0 | 0 | 23 |
| 3 ----- 5 | 50 | 0 | 0 | 50 |
| 3 ----- 6 | 11 | 0 | 0 | 11 |
| 3 ----- 7 | 4 | 0 | 0 | 4 |
| 3 ----- 8 | 0 | 0 | 0 | 0 |
| 3 ----- 9 | 42 | 0 | 0 | 42 |
| 3 ----- 10 | 11 | 0 | 0 | 11 |
| 4 ----- 11 | 23 | 0 | 0 | 0 |
| 4 -----141 | 169 | 0 | 0 | 146 |
| 5 ----- 23 | 50 | 0 | 0 | 0 |
| 6 ----- 24 | 11 | 0 | 0 | 0 |
| 7 ----- 25 | 4 | 0 | 0 | 0 |
| 8 ----- 26 | 0 | 0 | 0 | 0 |
| 9 ----- 27 | 42 | 0 | 0 | 0 |
| 10 ----- 28 | 11 | 0 | 0 | 0 |
| 11 ----- 12 | 23 | 0 | 0 | 0 |
| 13 ----- 16 | 2 | 0 | 0 | 0 |
| 14 ----- 15 | 11 | 0 | 0 | 0 |
| 15 ----- 17 | 12 | 0 | 0 | 1 |
| 16 ----- 18 | 2 | 2 | 0 | 0 |
| 17 ----- 19 | 13 | 0 | 0 | 1 |
| 18 ----- 20 | 0 | 0 | 0 | 0 |
| 21 ----- 43 | 119 | 0 | 0 | 0 |
| 22 ----- 42 | 131 | 0 | 0 | 0 |
| 24 ----- 37 | 11 | 11 | 0 | 0 |
| 25 ----- 29 | 4 | 0 | 0 | 0 |
| 26 ----- 30 | 0 | 0 | 0 | 0 |
| 27 ----- 31 | 42 | 0 | 0 | 0 |
| 29 ----- 32 | 4 | 0 | 0 | 0 |
| 30 ----- 33 | 0 | 0 | 0 | 0 |
| 31 ----- 36 | 42 | 0 | 0 | 0 |
| 33 ----- 34 | 0 | 0 | 0 | 0 |
| 34 ----- 35 | 0 | 0 | 0 | 0 |
| 37 ----- 38 | 0 | 0 | 0 | 0 |
| 38 ----- 39 | 0 | 0 | 0 | 0 |
| 39 ----- 44 | 0 | 0 | 0 | 0 |
| 39 ----- 40 | 108 | 0 | 0 | 108 |
| 39 -----177 | 92 | 0 | 0 | 92 |
| 41 -----165 | 108 | 0 | 0 | 0 |
| 42 ----- 43 | 132 | 13 | 0 | 1 |
| 43 ----- 41 | 119 | 11 | 0 | 0 |
| 44 ----- 45 | 41 | 0 | 0 | 41 |
| 44 ----- 47 | 0 | 0 | 0 | 0 |
| 45 ----- 51 | 42 | 0 | 0 | 1 |
| 46 ----- 52 | 41 | 0 | 0 | 0 |
| 48 ----- 53 | 0 | 0 | 0 | 0 |
| 50 ----- 54 | 11 | 0 | 0 | 0 |
| 53 ----- 61 | 0 | 0 | 0 | 0 |
| 54 ----- 58 | 11 | 0 | 0 | 0 |
| 55 ----- 65 | 41 | 0 | 0 | 0 |
| 56 ----- 63 | 44 | 0 | 0 | 0 |
| 57 ----- 59 | 12 | 0 | 0 | 0 |
| 58 ----- 74 | 11 | 0 | 0 | 0 |
| 60 ----- 72 | 13 | 0 | 0 | 0 |
| 62 ----- 71 | 0 | 0 | 0 | 0 |
| 64 ----- 69 | 47 | 0 | 0 | 0 |
| 66 ----- 67 | 44 | 0 | 0 | 0 |
| 68 ----- 75 | 47 | 0 | 0 | 0 |

| | | | | |
|---|---|---|---|---|
| 70 ----- 76 | 50 | 0 | 0 | 0 |
| 71 ----- 77 | 15 | 0 | 0 | 15 |
| 73 ----- 78 | 0 | 0 | 0 | 0 |
| 74 ----- 79 | 11 | 11 | 0 | 0 |
| 76 -----105 | 50 | 50 | 0 | 0 |
| 78 ----- 81 | 0 | 0 | 0 | 0 |
| 79 ----- 80 | 0 | 0 | 0 | 0 |
| 80 ----- 83 | 0 | 0 | 0 | 0 |
| 81 ----- 82 | 76 | 0 | 0 | 76 |
| 82 -----164 | 76 | 0 | 0 | 0 |
| 84 ----- 85 | 0 | 0 | 0 | 0 |
| 86 ----- 88 | 22 | 0 | 0 | 0 |
| 87 ----- 89 | 0 | 0 | 0 | 0 |
| 88 ----- 92 | 24 | 0 | 0 | 2 |
| 90 ----- 94 | 15 | 0 | 0 | 0 |
| 91 ----- 97 | 0 | 0 | 0 | 0 |
| 92 ----- 93 | 27 | 0 | 0 | 3 |
| 93 -----113 | 27 | 0 | 0 | 0 |
| 95 ----- 96 | 15 | 0 | 0 | 0 |
| 98 -----101 | 10 | 0 | 0 | 0 |
| 99 -----103 | 0 | 0 | 0 | 0 |
| 100 -----111 | 15 | 0 | 0 | 0 |
| 102 -----109 | 10 | 0 | 0 | 0 |
| 104 -----107 | 5 | 0 | 0 | 0 |
| 105 -----106 | 0 | 0 | 0 | 0 |
| 106 -----122 | 0 | 0 | 0 | 0 |
| 108 -----120 | 5 | 0 | 0 | 0 |
| 110 -----118 | 10 | 0 | 0 | 0 |
| 112 -----116 | 15 | 0 | 0 | 0 |
| 114 -----115 | 48 | 0 | 0 | 0 |
| 115 -----124 | 48 | 17 | 0 | 0 |
| 117 -----123 | 15 | 0 | 0 | 0 |
| 119 -----125 | 10 | 0 | 0 | 0 |
| 121 -----127 | 5 | 0 | 0 | 0 |
| 122 -----129 | 0 | 0 | 0 | 0 |
| 124 -----131 | 31 | 10 | 0 | 0 |
| 126 -----130 | 10 | 0 | 0 | 0 |
| 128 -----132 | 5 | 0 | 0 | 0 |
| 129 -----134 | 0 | 0 | 0 | 0 |
| 131 -----136 | 21 | 10 | 0 | 0 |
| 133 -----135 | 5 | 0 | 0 | 0 |
| 134 -----137 | 0 | 0 | 0 | 0 |
| 136 -----138 | 11 | 11 | 0 | 0 |
| 137 -----138 | 0 | 0 | 0 | 0 |
| 138 -----253 | 0 | 0 | 0 | 0 |
| 140 -----150 | 60 | 0 | 0 | 0 |
| 141 -----142 | 169 | 0 | 0 | 0 |
| 142 -----143 | 169 | 0 | 0 | 0 |
| 144 -----145 | 77 | 0 | 0 | 0 |
| 145 -----146 | 77 | 0 | 0 | 0 |
| 146 -----147 | 77 | 0 | 0 | 0 |
| 147 -----148 | 77 | 0 | 0 | 0 |
| 148 -----149 | 77 | 0 | 0 | 0 |
| 149 -----253 | 77 | 77 | 0 | 0 |
| 151 -----153 | 60 | 0 | 0 | 0 |
| 152 -----153 | 80 | 20 | 0 | 0 |
| 153 -----253 | 60 | 60 | 0 | 0 |
| 154 -----155 | 82 | 0 | 0 | 0 |
| 155 -----156 | 82 | 0 | 0 | 0 |
| 156 -----157 | 82 | 0 | 0 | 0 |

| | | | | | |
|---|---|---|---|---|---|
| 158 | -----159 | 82 | 0 | 0 | 0 |
| 159 | -----160 | 82 | 0 | 0 | 0 |
| 160 | -----161 | 82 | 0 | 0 | 0 |
| 161 | -----163 | 82 | 0 | 0 | 0 |
| 162 | -----161 | 82 | 0 | 0 | 0 |
| 163 | -----253 | 82 | 82 | 0 | 0 |
| 164 | -----168 | 76 | 0 | 0 | 0 |
| 166 | -----167 | 108 | 0 | 0 | 0 |
| 167 | -----174 | 122 | 6 | 0 | 14 |
| 167 | -----175 | 108 | 0 | 0 | 0 |
| 168 | -----170 | 80 | 0 | 0 | 4 |
| 168 | -----169 | 76 | 0 | 0 | 0 |
| 168 | -----171 | 76 | 0 | 0 | 0 |
| 172 | -----231 | 64 | 0 | 0 | 0 |
| 174 | -----205 | 116 | 52 | 0 | 0 |
| 175 | -----173 | 108 | 0 | 0 | 0 |
| 176 | -----193 | 104 | 0 | 0 | 0 |
| 177 | -----178 | 92 | 0 | 0 | 0 |
| 178 | -----179 | 106 | 0 | 0 | 14 |
| 178 | -----180 | 104 | 0 | 0 | 12 |
| 178 | -----181 | 92 | 0 | 0 | 0 |
| 178 | -----182 | 92 | 0 | 0 | 0 |
| 181 | -----185 | 97 | 0 | 0 | 5 |
| 182 | -----192 | 104 | 0 | 0 | 12 |
| 183 | -----187 | 104 | 0 | 0 | 0 |
| 184 | -----186 | 92 | 0 | 0 | 0 |
| 188 | -----195 | 92 | 0 | 0 | 0 |
| 189 | -----197 | 63 | 0 | 0 | 0 |
| 190 | -----201 | 107 | 43 | 0 | 0 |
| 191 | -----194 | 100 | 5 | 0 | 0 |
| 194 | -----202 | 95 | 0 | 0 | 0 |
| 196 | -----199 | 63 | 0 | 0 | 0 |
| 197 | -----198 | 65 | 0 | 0 | 2 |
| 198 | -----209 | 92 | 0 | 0 | 27 |
| 200 | -----208 | 63 | 0 | 0 | 0 |
| 201 | -----206 | 64 | 0 | 0 | 0 |
| 203 | -----204 | 104 | 9 | 0 | 0 |
| 204 | -----205 | 95 | 31 | 0 | 0 |
| 205 | -----230 | 64 | 0 | 0 | 0 |
| 207 | -----250 | 79 | 0 | 0 | 0 |
| 208 | -----220 | 63 | 0 | 0 | 0 |
| 209 | -----210 | 97 | 0 | 0 | 5 |
| 209 | -----211 | 92 | 0 | 0 | 0 |
| 209 | -----213 | 102 | 0 | 0 | 10 |
| 210 | -----217 | 99 | 0 | 0 | 2 |
| 211 | -----215 | 92 | 0 | 0 | 0 |
| 211 | -----214 | 94 | 0 | 0 | 2 |
| 212 | -----216 | 92 | 0 | 0 | 0 |
| 218 | -----219 | 92 | 0 | 0 | 0 |
| 219 | -----253 | 92 | 92 | 0 | 0 |
| 220 | -----221 | 71 | 0 | 0 | 8 |
| 220 | -----222 | 68 | 0 | 0 | 5 |
| 220 | -----223 | 63 | 0 | 0 | 0 |
| 224 | -----225 | 63 | 0 | 0 | 0 |
| 225 | -----226 | 63 | 0 | 0 | 0 |
| 226 | -----227 | 63 | 0 | 0 | 0 |
| 227 | -----253 | 63 | 63 | 0 | 0 |
| 228 | -----233 | 91 | 0 | 0 | 0 |
| 229 | -----232 | 79 | 0 | 0 | 0 |

| | | | | | |
|---|---|---|---|---|---|
| 231 | -----239 | 64 | 0 | 0 | 0 |
| 232 | -----238 | 79 | 0 | 0 | 0 |
| 233 | -----234 | 91 | 0 | 0 | 0 |
| 233 | -----235 | 93 | 0 | 0 | 2 |
| 233 | -----236 | 92 | 0 | 0 | 1 |
| 237 | -----251 | 91 | 0 | 0 | 0 |
| 238 | -----243 | 90 | 0 | 0 | 11 |
| 238 | -----244 | 90 | 0 | 0 | 11 |
| 238 | -----241 | 79 | 0 | 0 | 0 |
| 239 | -----240 | 64 | 0 | 0 | 0 |
| 240 | -----246 | 64 | 0 | 0 | 0 |
| 241 | -----242 | 79 | 0 | 0 | 0 |
| 242 | -----245 | 79 | 0 | 0 | 0 |
| 245 | -----247 | 79 | 0 | 0 | 0 |
| 246 | -----248 | 64 | 0 | 0 | 0 |
| 247 | -----253 | 79 | 79 | 0 | 0 |
| 248 | -----253 | 64 | 64 | 0 | 0 |
| 249 | ----- 49 | 11 | 0 | 0 | 0 |
| 251 | -----252 | 91 | 0 | 0 | 0 |
| 252 | -----253 | 91 | 91 | 0 | 0 |
| 253 | -----254 | 0 | 0 | 0 | 0 |

IN STMT 10 PROGRAM RETURNS FROM MAIN PROCEDURE.

IN STMT 10 SCALARS AND BLOCK-TRACE:

***** MAIN PROCEDURE DFSCRIP

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| L= | 0 | Z= | 10 | Y= | 3 | X= | 253 | P= | ? |
| M= | 1 | K= | 0 | J= | 254 | I= | 255 | C= | 367 |
| B= | 0 | A= | 732 | TOTAL_EVENTS= | 254 | | | | |

NON-0 PROCEDURE EXECUTION COUNTS:

| NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CP_ROUTE | 0210 | 00065 | CRITICALPAT | 0171 | 00734 | BUILD_TREE | 0134 | 00734 | TRACE | 0120 | 00732 | SEARCH | 0014 | 00001 |
| DFSCRIP | 0001 | 00001 | | | | | | | | | | | | |

LABEL EXECUTION COUNTS:

| NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| NEXT | 0115 | 00366 | OUT | 0066 | 00253 | JUMP | 0057 | 00366 | LOOP | 0037 | 00366 | | | |

| COMPILATION STATISTICS (0244 STATEMENTS) | | | | | | | EXECUTION STATISTICS | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SECONDS | ERRORS | WARNINGS | PAGES | LINES | CARDS | INCL'S | SECONDS | ERRORS | WARNINGS | PAGES | LINES | CARDS | INCL'S | AUX I/O |
| .15 | 0 | 1 | 0 | 4 | 468 | 0 | .68 | 0 | 0 | 10 | 481 | 254 | 0 | 0 |

| BYTES | SYMBOL TABLE | INTERMEDIATE CODE | OBJECT CODE | STATIC CORE | AUTOMATIC CORE | DYNAMIC CORE | TOTAL STORAGE |
|---|---|---|---|---|---|---|---|
| USED | 8670( 9K) | 6560( 7K) | 12448( 13K) | 342( 1K) | 89982( 88K) | 0( 0K) | 110646( 109K) |
| UNUSED | 8130( 105K) | 267767( 261K) | 371372( 362K) | 281038( 274K) | 281038( 274K) | 267767( 261K) | 281038( 274K) |

THIS PROGRAM MAY BE RERUN WITHOUT CHANGE IN A REGION 274K BYTES SMALLER USING TABLESIZE= 2168

APPENDIX 507

DFS Run, No Activities

```
IN STMT    10  SCALARS AND BLOCK-TRACE:

***** MAIN PROCEDURE DFSCRIP

L=          0          Z=       0          Y=       2            X=       1            P=       ?
M=          1          K=       0          J=       3            I=       3            C=       2
B=          0          A=       2          TOTAL_EVENTS=       2


NON-O PROCEDURE EXECUTION COUNTS:
NAME         STMT COUNT  NAME        STMT COUNT  NAME        STMT COUNT  NAME        STMT COUNT  NAME        STMT COUNT
CP_ROUTE     0210 00002  CRITICALPAT 0171 00004  BUILD_TREE  0134 00004  TRACE       0120 00002  SEARCH      0014 00001
DFSCRIP      0001 00001

LABEL EXECUTION COUNTS:
NAME         STMT COUNT  NAME        STMT COUNT  NAME        STMT COUNT  NAME        STMT COUNT  NAME        STMT COUNT
NEXT         0115 00001  OUT         0066 00001  JUMP        0057 00001  LOOP        0037 00001


       COMPILATION STATISTICS  (0244 STATEMENTS)       |                     EXECUTION STATISTICS
SECONDS  ERRORS  WARNINGS  PAGES  LINES  CARDS  INCL'S | SECONDS  ERRORS  WARNINGS  PAGES  LINES  CARDS  INCL'S  AUX I/O
   .15       0         1      0      4    468       0 |     .00       0         0      3     20      2       0        0
-------------------------------------------------------+---------------------------------------------------------------
BYTES   SYMBOL TABLE   INTERMEDIATE CODE   OBJECT CODE |   STATIC CORE   AUTOMATIC CORE    DYNAMIC CORE   TOTAL STORAGE
USED     8670(   9K)       6560(   7K)    12448(  13K) |     342(   1K)     3678(   4K)        0(  0K)    24342(  24K)
UNUSED  33458(  32K)      35239(  34K)    64172(  62K) |   60142(  58K)    60142(  58K)    35239( 34K)    60142(  58K)

THIS PROGRAM MAY BE RERUN WITHOUT CHANGE IN A REGION  58K BYTES SMALLER USING TABLESIZE=  2168
```

APPENDIX 508

Conventional Run, No Activities

IN STMT 10 SCALARS AND BLOCK-TRACE:

***** MAIN PROCEDURE CONCRIP

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| L= | ? | Z= | 0 | Y= | 2 | X= | 1 | P= | ? |
| M= | 1 | K= | 0 | J= | 0 | I= | 3 | C= | ? |
| B= | 0 | A= | 2 | TOTAL_EVENTS= | 2 | | | | |

NON-0 PROCEDURE EXECUTION COUNTS:

| NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PASS | 0120 | 00002 | TRACE | 0107 | 00002 | EXECUTE | 0014 | 00001 | CONCRIP | 0001 | 00001 | | | |

LABEL EXECUTION COUNTS:

| NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT | NAME | STMT | COUNT |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| AHEAD | 0201 | 00000 | CONTINUE | 0186 | 00000 | FURTHER | 0182 | 00000 | HOP | 0166 | 00001 | SKIP | 0149 | 00001 |
| BACK | 0132 | 00002 | JUMP | 0105 | 00001 | NEXT | 0101 | 00001 | OUT | 0061 | 00001 | LOOP | 0033 | 00001 |

| COMPILATION STATISTICS (0216 STATEMENTS) | | | | | | | EXECUTION STATISTICS | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SECONDS | ERRORS | WARNINGS | PAGES | LINES | CARDS | INCL'S | SECONDS | ERRORS | WARNINGS | PAGES | LINES | CARDS | INCL'S | AUX I/O |
| .12 | 0 | 1 | 0 | 4 | 401 | 0 | .01 | 0 | 0 | 3 | 29 | 2 | 0 | 0 |

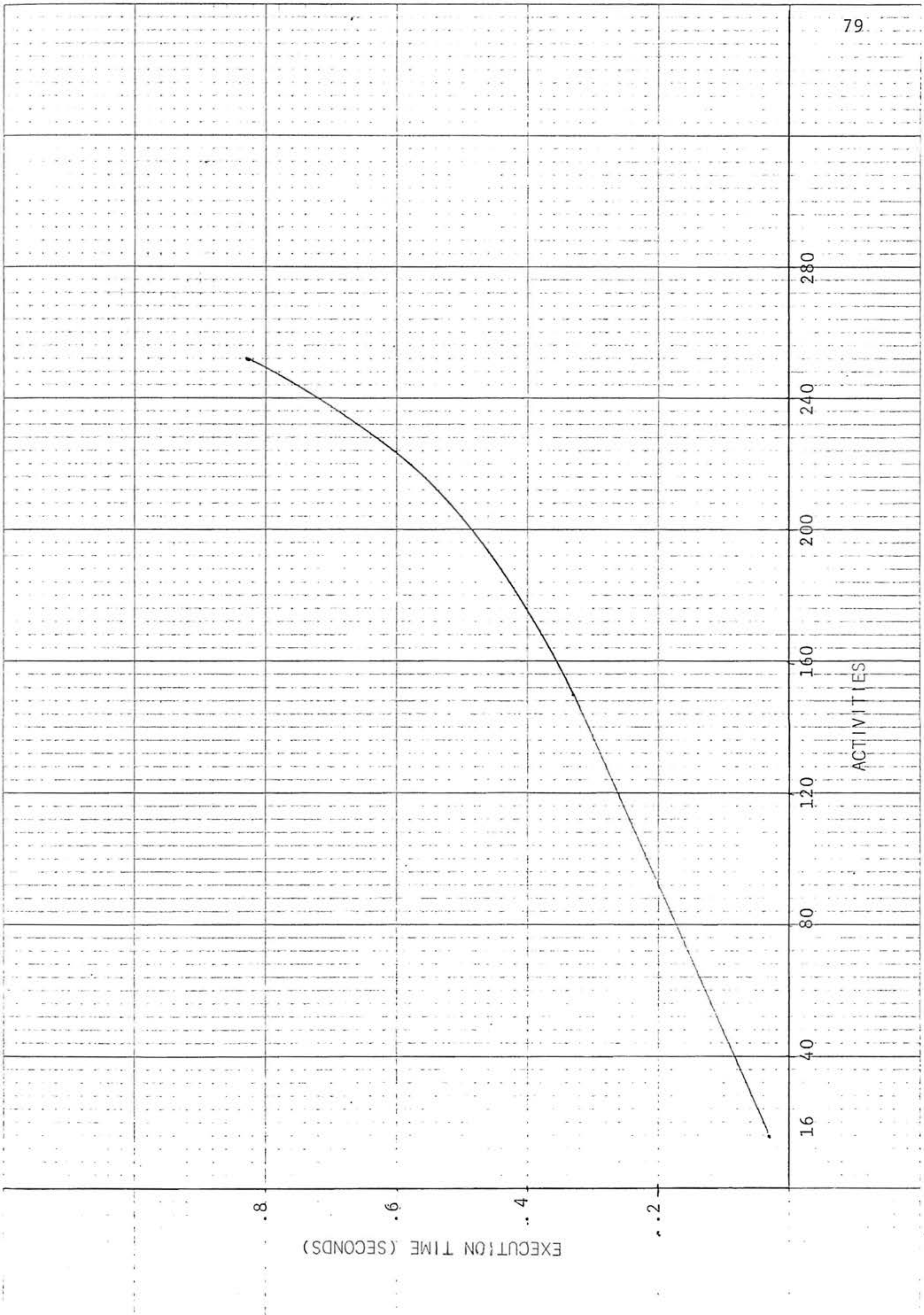
| BYTES | SYMBOL TABLE | INTERMEDIATE CODE | OBJECT CODE | STATIC CORE | AUTOMATIC CORE | DYNAMIC CORE | TOTAL STORAGE |
|---|---|---|---|---|---|---|---|
| USED | 8199( 9K) | 5638( 6K) | 10656( 11K) | 342( 1K) | 4742( 5K) | 0( 0K) | 23158( 23K) |
| UNUSED | 33929( 33K) | 36161( 35K) | 66418( 64K) | 61326( 59K) | 61326( 59K) | 36161( 35K) | 61326( 59K) |

THIS PROGRAM MAY BE RERUN WITHOUT CHANGE IN A REGION 59K BYTES SMALLER USING TABLESIZE= 2050

APPENDIX 509

Graph of Executive Time vs. Activities,

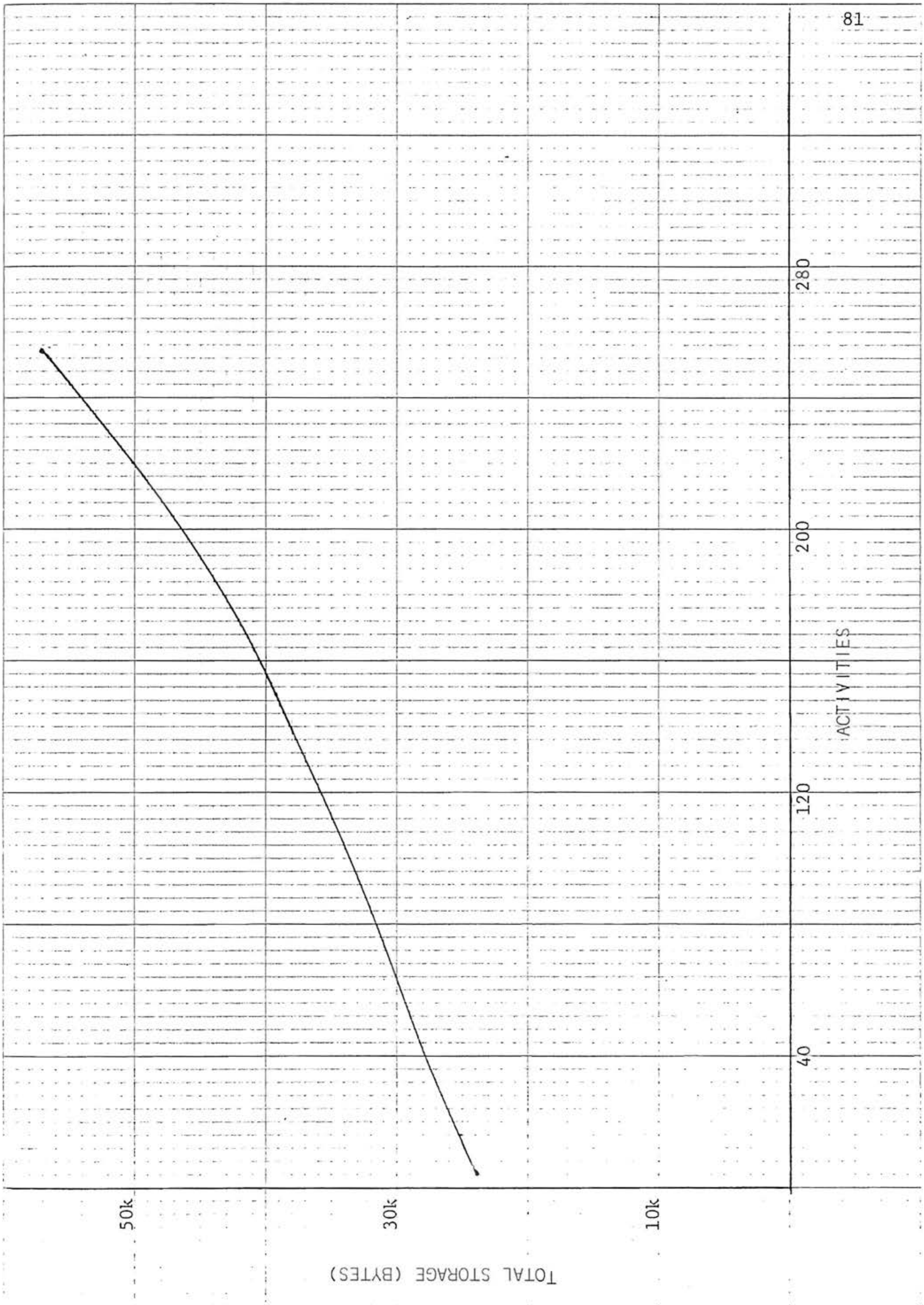
Conventional Method


ACTIVITIES
16
40
80
120
160
200
240
280
EXECUTION TIME (SECONDS)
.2
.4
.6
.8

APPENDIX 510

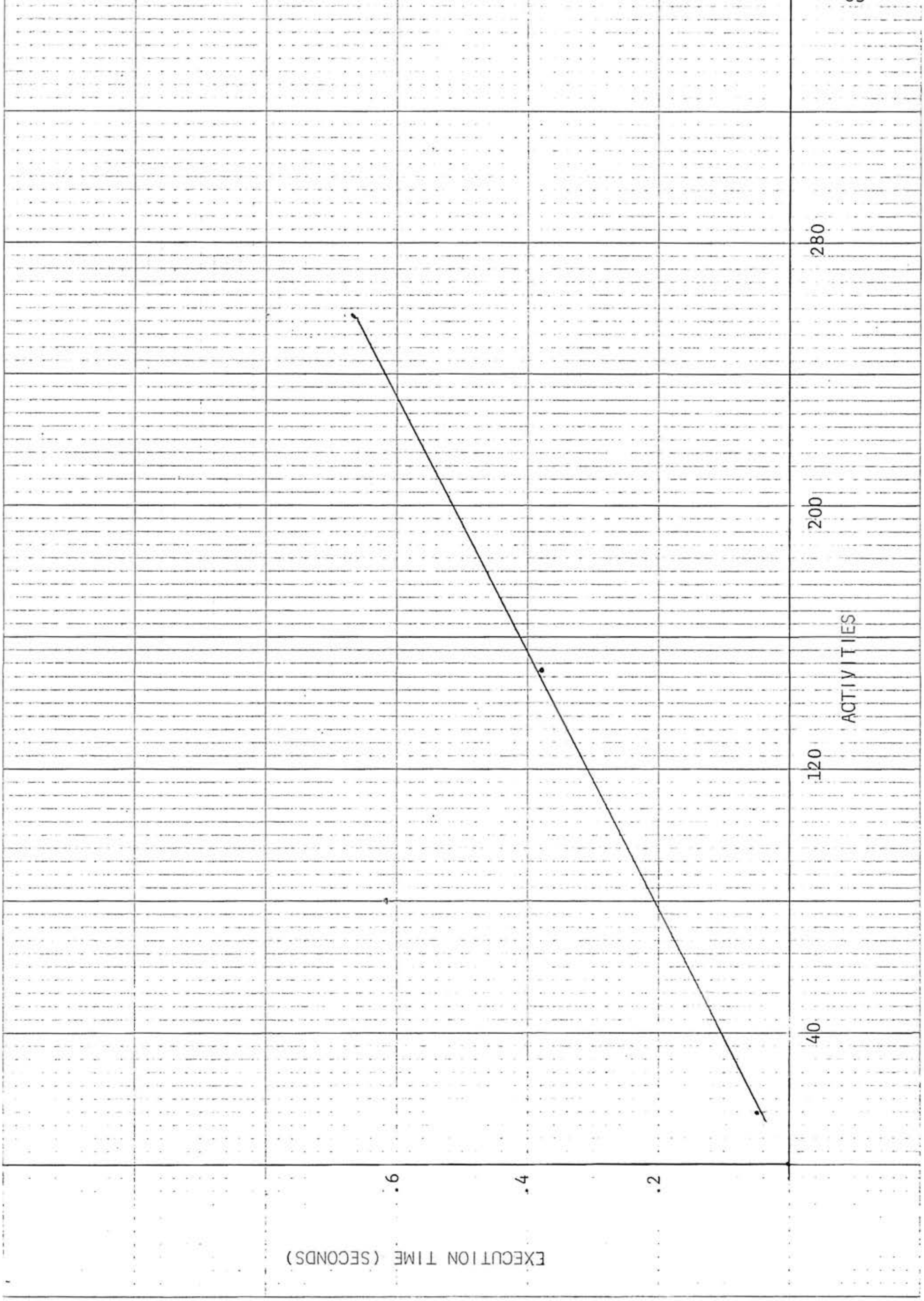
Graph of Total Storage vs. Activities

Conventional Method


ACTIVITIES
40
120
200
280
TOTAL STORAGE (BYTES)
50k
30k
10k

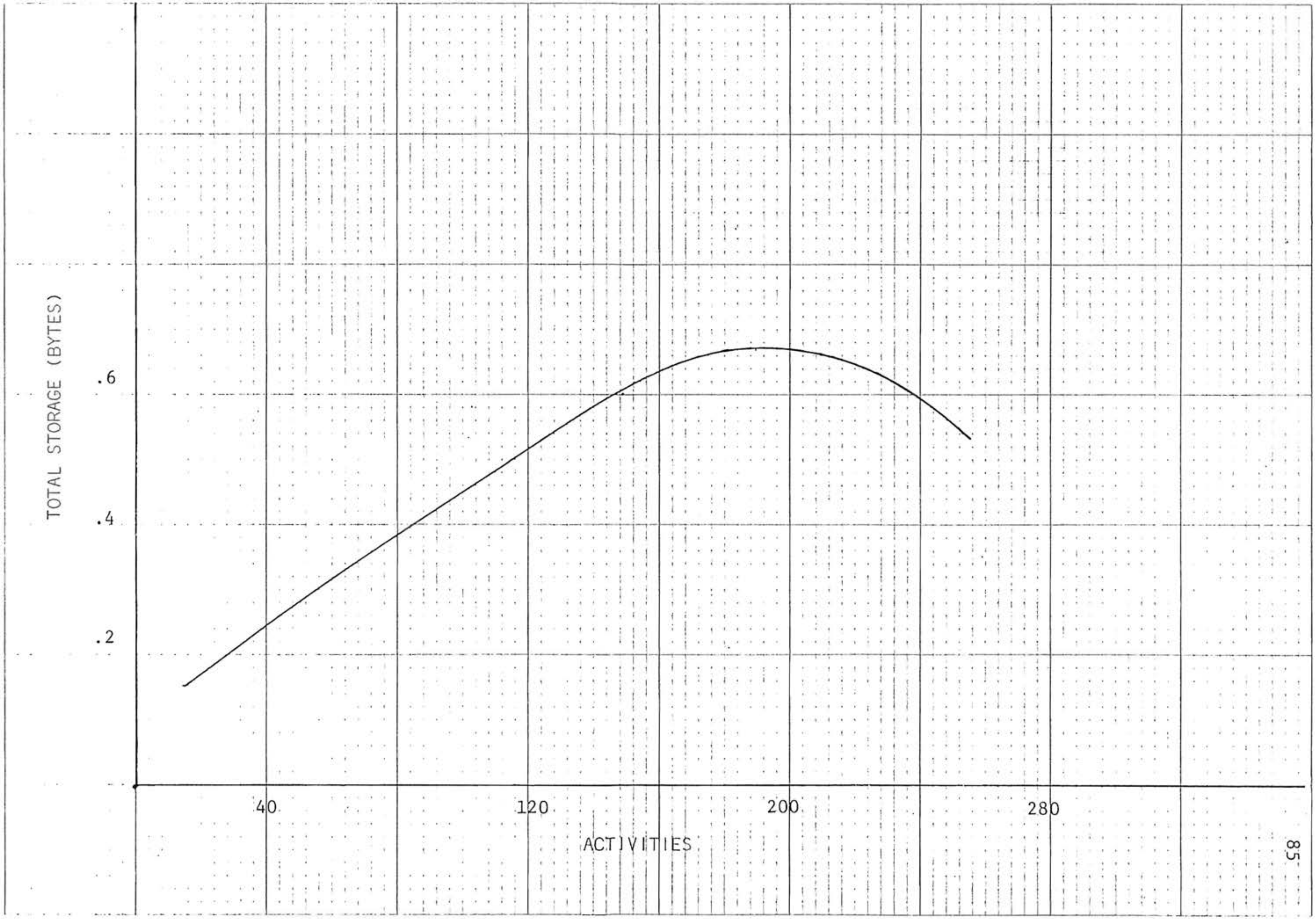
APPENDIX 511

Graph of Executive Time vs. Activities, DFS


ACTIVITIES
40
120
200
280
EXECUTION TIME (SECONDS)
.2
.4
.6

APPENDIX 512

Graph of Total Storage vs. Activities, DFS


TOTAL STORAGE (BYTES)
.6
.4
.2
40
120
200
280
ACTIVITIES

VITA

Jeffrey A. Scofield

Candidate for the Degree of

Master of Business Administration

Report: EVALUATION OF A DEPTH-FIELD SEARCH ALGORITHM FOR CRITICAL PATH METHOD

Major Field: Business Administration

Biographical:

Personal Data: Born in Penn Yan, New York, February 13, 1958, the son of George and Hilda Scofield.

Education: Graduated from Penn Yan Academy, May 1976; while serving in the United States Air Force earned a total of 78 credit hours by attending classes part time at Arkansas State University and the University of Little Rock; received the Bachelor of Science degree from Oklahoma State University with a major in Electronics Engineering Technology in December 1982; completed requirements for the Master of Business Administration at Oklahoma State University, July 1984.

Professional Experience: Graduate assistant to a professor of management in the College of Business Administration, Oklahoma State University. January 1983 to May 1984; electronic design engineer, Oklahoma State University, August 1982 to December 1982; electronics technician, Physics Department, Oklahoma State University, October 1981 to April 1982; transportation construction inspector, Department of Transportation, New York, summer 1981; missile systems analyst technician, United States Air Force, August 1976 to August 1980.