(2)

NAVAL POSTGRADUATE SCHOOL
Monterey, California

DTIC
ELECTE
OCT. 10 1991
S B D

THESIS

APPROXIMATE INTERVAL ESTIMATION
METHODS FOR THE RELIABILITY OF
SYSTEMS USING DISCRETE
COMPONENT DATA

by

Edmundo F. Bellini

September 1990

Thesis Advisor W. Max Woods

Unclassified

security classification of this page

REPORT DOCUMENTATION PAGE

1a Report Security Classification Unclassified	1b Restrictive Markings
2a Security Classification Authority	3 Distribution Availability of Report
2b Declassification Downgrading Schedule	Approved for public release; distribution is unlimited.
4 Performing Organization Report Number(s)	5 Monitoring Organization Report Number(s)
6a Name of Performing Organization Naval Postgraduate School — 6b Office Symbol *(if applicable)* 55Wo	7a Name of Monitoring Organization Naval Postgraduate School
6c Address *(city, state, and ZIP code)* Monterey, CA 93943-5000	7b Address *(city, state, and ZIP code)* Monterey, CA 93943-5000
8a Name of Funding Sponsoring Organization — 8b Office Symbol *(if applicable)*	9 Procurement Instrument Identification Number
8c Address *(city, state, and ZIP code)*	10 Source of Funding Numbers: Program Element No / Project No / Task No / Work Unit Accession No

11 Title *(include security classification)* APPROXIMATE INTERVAL ESTIMATION METHODS FOR THE RELIABILITY OF SYSTEMS USING DISCRETE COMPONENT DATA

12 Personal Author(s) Edmundo F. Bellini

13a Type of Report	13b Time Covered	14 Date of Report *(year, month, day)*	15 Page Count
Master's Thesis	From To	September 1990	83

16 Supplementary Notation The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17 Cosati Codes			18 Subject Terms *(continue on reverse if necessary and identify by block number)*
Field	Group	Subgroup	Binomial, System Reliability, Chi-Square statistic,

19 Abstract *(continue on reverse if necessary and identify by block number)*

Three lower confidence interval estimation procedures for system reliability of coherent systems with cyclic components are developed and their accuracy evaluated by Monte Carlo methods. Each method uses estimates of the ratios of component unreliabilities and the Poisson approximation to the binomial distribution to obtain the equation for the lower confidence limit. This is an extension of a method previously reported in the literature which has been shown to be fairly robust. The procedures developed here can be combined with similar procedures already developed for systems with continuous components. The combined procedure may yield a reasonably accurate lower confidence interval procedure for the reliability of coherent systems with mixtures of continuous and cyclic components.

20 Distribution Availability of Abstract ☒ unclassified unlimited ☐ same as report ☐ DTIC users	21 Abstract Security Classification Unclassified	
22a Name of Responsible Individual W. Max Woods	22b Telephone *(include Area code)* (408) 646-2786	22c Office Symbol 55Wo

DD FORM 1473,84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete — security classification of this page

Unclassified

Approved for public release; distribution is unlimited.

Approximate Interval Estimation Methods for the Reliability of Systems Using Discrete Component Data

by

Edmundo F. Bellini
Lieutenant, United States Navy
B.S., United States Air Force Academy, 1976

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 1990

Author: Edmundo F. Bellini

Edmundo F. Bellini

Approved by: W. M. Woods

W. Max Woods, Thesis Advisor

H. J. Larson

Harold J. Larson, Second Reader

P. Purdue

Peter Purdue, Chairman,
Department of Operations Research

ABSTRACT

Three lower confidence interval estimation procedures for system reliability of coherent systems with cyclic components are developed and their accuracy evaluated by Monte Carlo methods. Each method uses estimates of the ratios of component unreliabilities and the Poisson approximation to the binomial distribution to obtain the equation for the lower confidence limit. This is an extension of a method previously reported in the literature which has been shown to be fairly robust. The procedures developed here can be combined with similar procedures already developed for systems with continuous components. The combined procedure may yield a reasonably accurate lower confidence interval procedure for the reliability of coherent systems with mixtures of continuous and cyclic components.

THESIS DISCLAIMER

The reader is cautioned that computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made, within the time available, to ensure that the programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at the risk of the user.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. BACKGROUND

A coherent system is any system whose reliability is not reduced when the reliability of any of its components is increased. This thesis deals with coherent systems that have cyclic components. Cyclic components perform one or more repetitions of a single function during their mission. Examples are components that have power-turn-ons and power-turn-offs, switches and hydraulic units. Any component that performs a function which is regarded as a success or failure is a cyclic component. Consequently a continuously operating electronic component becomes a cyclic component if the only data for that component is the number of successes and failures in repeated trials. The number of trials to first failure of cyclic components have discrete probability distributions. Numerous approximate interval estimation procedures have been developed for system reliability of coherent systems with cyclic components. Such a method developed by Myhre and Saunders [Ref. 1: p.37] uses a likelihood ratio method. C. R. Rao [Ref. 2] developed a maximum likelihood method and Easterling [Ref. 3] proposed a modified maximum likelihood method. For special structures, such as series systems, exact methods were developed by Winterbottom [Ref. 4: pp.782-787]. Other approximate and asymptotic methods were developed by Mann and Grubbs [Ref. 5]. The accuracy of some of these approximate procedures has been studied for specific sample size cases for structures of order two or three by Mann and others [Ref. 6] and Winterbottom [Ref. 4: pp.782-787]. Woods and Borsting [Ref. 7] developed an approximate procedure which was modified extensively by Mann and Grubbs [Ref. 5: pp.335-347]. None of these discrete interval estimation procedures cited above can be readily used in conjunction with continuous component data to obtain interval estimates for the reliability of complex systems that have mixtures of cyclic and continuous components.

In this thesis we attempt to establish feasible interval estimation methods for the reliability of coherent systems with cyclic components. These methods have a common feature that allow them to be combined with similar methods that use continuous data. The combination of these methods may provide interval estimates for the reliability of systems with cyclic and continuous components. This method is an extension of a method developed by Lomnicki [Ref. 8] and extended by Myhre and others [Ref. 9: p.213]. They compare the accuracy of their procedure with the accuracy of three of the procedures cited above and provide bounds for error between the true lower confidence bound and their approximate lower bound. These bounds are used to show that their procedure is fairly robust against errors in assumptions about the ratios between component unreliabilities which are needed to use their procedure. The procedure developed in this thesis attempts to capitalize on this robustness to extend their method by *estimating these ratios from the test data* . Specifically, let R_i denote the reliability of component i and denote component unreliability by $q_i = 1 - R_i$. Suppose the component unreliabilities of a coherent system of size k are denoted by $q_1, q_2, \dots, q_k$. Let

$$q_m = \max\{q_1, q_2, \dots, q_k\} \tag{1.1}$$

and $a_i = \frac{q_i}{q_m}$. Then the reliability function $h(q_1, q_2, \dots, q_k)$, of the system can be written as $h(q_m, a_1, a_2, \dots, a_k)$. Since the system is coherent, a lower confidence limit on system reliability $\hat{R}_{S,L}$ is given by

$$\hat{R}_{S,L} = h(\hat{q}_u, a_1, \dots, a_k) \tag{1.2}$$

where $\hat{q}_u$ is the corresponding upper confidence limit on q_m. In this thesis the a_i are estimated from the data. The method developed by Myhre and others [Ref. 9: pp.216-223] assume the a_i are known.

Myhre and others [Ref. 9: p.223] also apply this failure rate ratio concept to systems whose components have exponentially distributed failure times with failure rates λ_i. In this case,

$$\lambda_m = \max\{\lambda_1, \lambda_2, \dots, \lambda_k\} \tag{1.3}$$

$$a_i = \frac{\lambda_i}{\lambda_m} \tag{1.4}$$

and $h(\hat{\lambda}_u, a_1, \dots, a_k)$ is the lower confidence limit for the reliability function $h(\lambda_m, a_1, \dots, a_k)$ where $\hat{\lambda}_u$ is the upper confidence limit for λ_m . In his thesis, Lee [Ref. 10: p.7] has demonstrated that if the same life data used to obtain $\hat{\lambda}_u$ is used to construct estimates $\hat{a}_i$ for the a_i, in equation (1.4), then the approximate lower confidence limit

$$h(\hat{\lambda}_u, \hat{a}_1, \hat{a}_2, \dots, \hat{a}_k) \tag{1.5}$$

is an accurate approximate interval estimation procedure.

A primary reason for investigating the accuracy of this failure rate ratio method for cyclic systems is that it can be easily extended to systems that have mixtures of cyclic and continuous components. If the procedure investigated here for systems of cyclic components is found to be accurate, then it can readily be combined with the procedures established by Lee [Ref. 10: pp.3-23] to obtain an accurate interval estimation procedure for the reliability of a coherent system with a mixture of cyclic and continuous components.

An additional primary purpose of this thesis is to develop an extensive computer simulation program that will provide a means for evaluating the accuracy of proposed interval estimation methods for the reliability of coherent systems with cyclic components.

II. INTERVAL ESTIMATION PROCEDURES FOR SYSTEM RELIABILITY

A. GENERAL CONCEPTS

Consider a coherent system with k cyclic components. Mission tests for component i consist of independent Bernoulli trials administered under a mission operating environment with probability q_i of failure on each test. Then F_i, the number of failures in n_i independent mission tests, has a binomial distribution denoted by $\mathrm{BIN}(n_i,q_i)$. We shall refer to q_i as the *unreliability* of component i.

Utilizing the concept first suggested by Lomnicki [Ref. 8: p. 109] and expanded by Myhre and others [Ref. 9: p.213], the unreliability of each component can be written as a fraction of the unreliability of the least reliable component; that is

$$q_i = a_i q_m \tag{2.1}$$

where

$q_m = \max_i\{q_i\}$, $\quad i = 1, 2, \dots, k \quad$ and $\quad 0 \le a_i \le 1$. We shall assume that the q_i are sufficiently small so that the probability distribution of F_i can be approximated by the Poisson distribution with mean $n_i q_i$. We denote this by the expression

$$F_i \sim P(n_i q_i). \tag{2.2}$$

Therefore, the distribution of the total number of failures, $F = \sum_{i=1}^{k} F_i$, can be approximated by the Poisson distribution with mean $\sum_{i=0}^{k} n_i q_i$; that is,

$$F \sim P\left(\sum_{i=1}^{k} n_i q_i\right). \tag{2.3}$$

Applying equation (2.1) to equation (2.3) we obtain

$$F \sim P\left(q_m \sum_{i=1}^{k} n_i a_i\right). \tag{2.4}$$

Let $h(q_1, q_2, \dots, q_k)$ denote the equation for system reliability, R_S. Then from equation (2.1),

$$R_S = \mathrm{h}(q_m, a_1, a_2, \dots, a_k). \tag{2.5}$$

Since the system is coherent, h is non-increasing in q_m. Therefore, a lower 100(1-α)% confidence limit, $R_{S,L(\alpha)}$ for R_S is

$$R_{S,L(\alpha)} = \mathrm{h}(\hat{q}_{m,u(\alpha)}, a_1, \dots, a_k) \tag{2.6}$$

where $\hat{q}_{m,u(\alpha)}$ is any 100(1-α)% upper confidence limit for q_m. To obtain $\hat{q}_{m,u(\alpha)}$ we use the following well known result for the upper confidence limit of the mean of a Poisson distribution, see [Ref. 11: p.218]. If $X \sim P(\lambda)$ then the upper 100(1-α)% confidence limit $\lambda_{u(\alpha)}$ for λ using one observation on X is

$$\lambda_{u(\alpha)} = \frac{\chi^2_{\alpha,2(1+X)}}{2}. \tag{2.7}$$

Applying this result to equation (2.4) we obtain the upper 100(1-α)% confidence limit for $q_m\sum_{i=1}^{k} n_i a_i$ which is given by

$$\left(q_m\sum_{i=1}^{k} n_i a_i\right)_{u(\alpha)} = \frac{\chi^2_{\alpha,2(1+F)}}{2}. \tag{2.8}$$

The corresponding upper confidence limit for q_m is

$$q_{m,u(\alpha)} = \frac{\chi^2_{\alpha,2(1+F)}}{2\sum_{i=1}^{k} n_i a_i}. \tag{2.9}$$

Equation (2.9) is the expression used by Myhre and others [Ref. 9 : p.214].

The components of the vector $\underline{a} = (a_1, a_2, \dots, a_k)$ as defined by Myhre and others [Ref. 9: p.214] are computed in relation to the largest unreliability q_m. However any component could be used as the base component to form a vector of ratios. Choosing the one with largest unreliability is a convenient way to select the index m. In this thesis we will assume the q_i are unknown. We will estimate them from the data. We also use the data to determine the base component, which we label component m, that will be used to form the ratio estimates $\hat{a}_i$. Specifically if N_i, F_i are the number of trials and resultant failures for component i, then $\hat{q}_i = \frac{F_i}{N_i}$ is the estimate of unreliability q_i. Hereafter in this thesis, we define the index m by the equation

$$\hat{q}_m = \max(\hat{q}_1, \hat{q}_2, \dots, \hat{q}_k). \tag{2.10}$$

We shall define $\hat{a}_i$ by

$$\hat{a}_i = \frac{\hat{q}_i}{\hat{q}_m} \qquad i = 1, 2, \dots, k. \tag{2.11}$$

Note that $\hat{q}_m$ is ***not intended*** to be an estimate of $\max(q_1, \dots, q_k)$. It is nothing more than the largest of the observed estimates of the component unreliabilities and provides us with the base-component which we denote by m. Hereafter, in this thesis, q_m will denote the unreliability of the component m which has been determined by the definition of $\hat{q}_m$. With the index m determined and the quantities $\hat{a}_1, \hat{a}_2, \dots, \hat{a}_k$ computed, we can compute an approximate upper confidence limit $\hat{q}_{m,u(\alpha)}$ for q_m that corresponds to $q_{m,u(\alpha)}$ in equation (2.9). Specifically,

$$\hat{q}_{m,u(\alpha)} = \frac{\chi^2_{\alpha,2(1+F)}}{2\sum_{i=1}^{k} n_i \hat{a}_i}. \tag{2.12}$$

The corresponding approximate lower 100(1-α)% confidence limit $\hat{R}_{S,L(\alpha)}$ is

$$h(\hat{q}_{m,u(\alpha)}, \hat{a}_1, \dots, \hat{a}_k). \tag{2.13}$$

We have special problems with equation (2.12) when none of the components fail. In this case $\hat{q}_i = 0$, $(i = 1, 2, \dots, k)$ $\hat{q}_m = 0$ and all $\hat{a}_i$ are undefined. If at least one component has at least one failure, equation (2.12) is well defined. Consequently, equations (2.12) and (2.13) must be modified or supplemented to account for cases when none or perhaps only one of the components fail. In fact we may want to modify the definition of $\hat{a}_i$ whenever $\hat{a}_i = 0$; i.e. when $F_i = 0$. Anytime $\hat{a}_i = 0$, the N_i tests for component i make no contribution toward the evaluation of $\hat{q}_{m,u(\alpha)}$ in equation (2.12). If information about the range of $a_i \equiv \frac{q_i}{q_m}$ is known, it could be used to redefine $\hat{a}_i$ so that $\hat{a}_i n_i$ in the denominator of equation (2.12) would not be zero. This is mathematically equivalent to adding $\hat{a}_i n_i$ tests to the N_m tests for component m without adding any failures to F_m. This is the one advantage of the ratio method and it is reflected in both equations (2.12) and (2.13). It is this specific property of the ratio method that makes it particularly appealing when the total number of failures is small.

The primary purpose of this thesis is to initiate the development of alternative methods for constructing approximate lower confidence limits $R_{S,L(\alpha)}$ for the reliability R_S of coherent systems that account for zero or nearly zero failures and to construct a computer program that can be used to evaluate the accuracy of these confidence limit procedures. It is highly desirable that these methods make strong use of the ratio method, when several types of components experience failure, because doing so should allow us to extend these confidence limit procedures to systems that have both cyclic and continuously operating types of components.

It is highly unlikely that any one confidence limit procedure will be reasonably accurate for all system configurations; i.e., for all reliability functions, $h(q_1, \ldots, q_k)$. Consequently, we have developed three similar procedures which are labeled Procedure 1, 2, and 3. Each of these procedures will be evaluated for accuracy when applied to two radically different types of system configurations; namely series system and Wheatstone bridge systems. Series systems have no redundant components. Wheatstone bridges are highly redundant systems. These accuracy evaluations should help establish some preliminary boundary constraints on the application of these procedures and provide insights for modifications that may yield more accurate procedures.

The first evaluations will be performed for a series system of k independent components. In this case the system reliability R_S is given by $R_S = \prod_{i=1}^{k}(1 - q_i) = \prod_{i=1}^{k}(1 - a_i q_m)$ where q_i denotes the unreliability of component i, and m will be the index established by the data as previously discussed, and $a_i = \dfrac{q_i}{q_m}$. One expression that can be used to construct a lower 100(1-α)% confidence limit $R_{S,L(\alpha)}$ for R_S using $\hat{q}_{m,u(\alpha)}$ given in equation (2.12) is

$$\hat{R}_{S,L(\alpha)} = \prod_{i=1}^{k}(1 - \hat{a}_i \, \hat{q}_{m,u(\alpha)}). \tag{2.14}$$

In the following descriptions, n_i and F_i denote the number of tests (Bernoulli trials) and the number of failures, respectively, for component i, $i = 1, 2, \ldots, k$.

B. PROCEDURE 1

This procedure has three expressions for $\hat{R}_{S,L(\alpha)}$ which depend on the test results $F_1, F_2, \ldots, F_k$. If no components fail, all $\hat{q}_i = 0$, and all $\hat{a}_i$ are undefined. This precludes the use of equation (2.12) to construct $\hat{R}_{S,L(\alpha)}$ for this case. When no components fail, we define

$$n^* = \min(n_1, n_2, \ldots, n_k) \tag{2.15}$$

and interpret the data as n^* successful system tests. Rather than use the standard approximate binomial confidence limit $(\alpha)^{\frac{1}{n^*}}$, we choose to define $\hat{R}_{S,L(\alpha)}$ by

$$\hat{R}_{S,L(\alpha)} = 1 - \frac{\chi^2_{\alpha,2}}{2n^*} \tag{2.16}$$

This expression reflects the Poisson approximation to the Binomial distribution and will be slightly conservative when all $F_i = 0$, but the other components to Procedure 1 are suspected to be slightly optimistic. So the entire procedure may be nearly exact. If the entire procedure is conservative, we can always change equation (2.16).

The second expression in Procedure 1 addresses the case when all but one component, say component 1, have zero failures. For a series system, this would correspond to at most one system failure out of n^* system tests. This would not be the case in other systems if the failed component is redundant in the system. Here again we choose slightly conservative procedure for the lower confidence limit; namely, we define $\hat{q}_{1,u(\alpha)}$ by

$$\hat{q}_{1,u(\alpha)} = \frac{\chi^2_{\alpha,2(1+F_1)}}{2n_1}. \tag{2.17}$$

We use equation (2.13) to define $\hat{R}_{S,L(\alpha)}$ for a general coherent system. If the system is a series system we use equation (2.14).

The third expression in Procedure 1 addresses the case when at least two different types of components have failures; that is, $F_i \neq 0$ for at least two values of i, $i = 1, 2, \ldots, k$.. In this case we use equation (2.12) to define $\hat{q}_{m,u(\alpha)}$ where $\hat{a}_i = \frac{\hat{q}_i}{\hat{q}_m}$, $\hat{q}_i = \frac{F_i}{n_i}$ and $\hat{q}_m = \max(\hat{q}_1, \ldots, \hat{q}_k)$. The lower confidence limit for the system is given by equation (2.13) for general coherent systems or equation (2.14) for a series system.

C. PROCEDURE 2

Procedure 2 differs from Procedure 1 in two respects. First, we assume that at least two different components of the system have at least one failure. Operationally this means that Procedure 2 would only be applied to data for which at least two types of system components experienced one or more failures. In this procedure we always use equation (2.1) to obtain $\hat{q}_{m,u(\alpha)}$ and equation (2.13) or (2.14) to compute $\hat{R}_{S,L(\alpha)}$ for general coherent systems or series systems respectively. Again $\hat{a}_i = \frac{\hat{q}_i}{\hat{q}_m}$, but we will define $\hat{q}_i$ differently for components with zero failures. Specifically, we apply a common scaling factor K to increase the sample size from n_i to Kn_i for all system components i which experienced zero failures and record 1 failure for each of those components. For example, if n_1, n_2 and n_3 tests were performed on component types 1, 2 and 3 and no failures occurred on any of these tests, the data is modified to show $F_1 = 1$, $F_2 = 1$, $F_3 = 1$ and new sample sizes Kn_1, Kn_2 and Kn_3. The scaling factor is an input parameter to the computer programs used to evaluate this procedure; ie., it is assumed to be known. It is determined by

$$K = \frac{\max(q_1, \dots, q_k)}{\min(q_1, \dots, q_k)} = \frac{q_{\max}}{q_{\min}} \tag{2.18}$$

where the q_i are the component unreliabilities. Operationally, this would mean that the user of this procedure must determine an estimate for this ratio. Previous data is often available to provide this estimate. Such estimates can sometimes be constructed from Department of Defense (DOD) documents that provide a variety of quality data characteristics for hardware purchased in accordance with prescribed standards and specifications; e.g., MILHDBK-217E and its referenced documents. Also the reliability values for components used in reliability prediction analysis are usually derived from some official source which could be satisfactory for such estimates. A reliability prediction analysis is usually required in any major DOD system acquisition program. If a satisfactory estimate for K cannot be obtained, then Procedure 2 should not be used. Of course one of the purposes of this thesis is to investigate the accuracy of this procedure when the scaling factor K is chosen "correctly"; i.e., according to equation (2.18). Thinking of q_{max} and q_{min} as failure rates, $\frac{1}{q_{max}}$ and $\frac{1}{q_{min}}$ denote the expected number of tests to first failure for their respective components. Then the value K in equation (2.18) is the ratio of these two expected number of tests to obtain one failure for each component. That is,

$$K = \left(\frac{1}{q_{\min}}\right) / \left(\frac{1}{q_{\max}}\right). \tag{2.19}$$

For each system component i that experiences zero failures, the corresponding estimate $\hat{q}_i$ for q_i is given by

$$\hat{q}_l = \frac{1}{K n_l} . \tag{2.20}$$

Then

$$\hat{a}_l = \frac{\hat{q}_l}{\hat{q}_m} \tag{2.21}$$

and

$$\hat{q}_{m,u(\alpha)} = \frac{\chi^2_{\alpha,\,2(1+F)}}{2 \sum_{i=0}^{k} n_l \, \hat{a}_l} \tag{2.22}$$

where F is the number of actual failures observed before the data is adjusted. Thus the data is adjusted to obtain estimated values $\hat{a}_l$ different from zero for those components that have no failures. The value for $\hat{R}_{S,L(\alpha)}$ is given by equation (2.14). Note that if all k types of components have at least one failure, the resulting value of $\hat{R}_{S,L(\alpha)}$ is the same under Procedures 1 and 2.

D. PROCEDURE 3

Procedure 3 differs from Procedure 1 in only one respect. We scale the sample sizes as we did in Procedure 2. That is, for each system component i that has no failures, we estimate q_l from equation (2.20). Equations for $\hat{a}_l$, $\hat{q}_{m,(\alpha)}$ and $\hat{R}_{S,L(\alpha)}$ are given by equations (2.21), (2.22) and (2.13) or (2.14) respectively.

Procedure 3 does not require that at least one failure have occurred on at least two different types of components as required in Procedure 2. Operationally, Procedure 3 can be used for all sets of data including those sets where no failures occur.

E. BRIDGE SYSTEM

We define a Wheatstone bridge system by the reliability block diagram shown in Figure 1 on page 11. If p_i denotes the reliability of component i, $i = 1, 2, \dots, 5$, and $1 - p_i \equiv q_i = a_i q_m$ where $q_m = \max\{q_1, q_2, \dots, q_5\}$, then system reliability R_S is given by

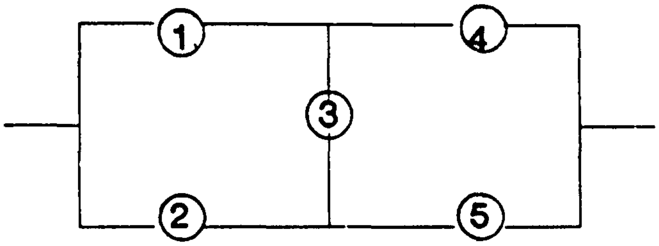

Figure 1. Block Diagram of a Wheatstone Bridge Structure.

$$
\begin{aligned}
R_S &= \mathrm{h}(q_m, a_1, \ldots, a_5) \\
&= 1 - q_m^2(a_1a_2 + a_4a_5) - q_m^3(a_1a_3a_5 + a_2a_3a_4) \\
&+ q_m^4(a_1a_2a_3a_4 + a_1a_2a_3a_5 + a_1a_2a_4a_5 + a_1a_3a_4a_5 + a_2a_3a_4a_5) \\
&- 2q_m^5(a_1a_2a_3a_4a_5)
\end{aligned} \tag{2.23}
$$

(see Myhre and others [Ref. 9: p.215]). Then the equation for the lower 100(1-α)% confidence limit on this system is

$$
\hat{R}_{S,\,L(\alpha)} = \mathrm{h}(\hat{q}_{m,\,u(\alpha)}, \hat{a}_1, \ldots, a_5) \tag{2.24}
$$

where $\hat{q}_{m,\,u(\alpha)}$ and $\hat{a}_i$ are defined as in Procedures 1, 2 or 3. Only Procedure 1 was evaluated for the bridge system and reported in this thesis.

III. SIMULATION PROCEDURE

Standard computer simulation methods were used to evaluate the accuracy of the three interval estimation methods discussed in section II. The input parameters needed to run the computer programs for each of the three interval estimation procedures are as follows:

- k: number of components in the system
- $\underline{n} = (n_1, n_2, \dots, n_k)$: vector of sample sizes(mission tests)
- $\underline{q} = (q_1, q_2, \dots, q_k)$: vector of component unreliabilities
- α: level of confidence

The NON-IMSL uniform random number generator SRND was used to simulate the outcomes of mission tests on the k components. A total of $\sum_{i=1}^{k} n_i$ uniform random numbers were generated and transformed into ones or zeroes as follows: the n_i numbers $x_{i1}, x_{i2}, \dots, x_{in_i}$ in the ith block of uniform random numbers were transformed by the expression

$$Y_{ij} = 1 \quad \textit{if } x_{ij} \leq q_i$$

$$Y_{ij} = 0 \quad \textit{if } x_{ij} > q_i$$

for $j = 1, 2, \dots, n_i$, and $i = 1, 2, \dots, k$, where q_i is the input parameter denoting the unreliability of component i. Then $F_i = \sum_j Y_{ij}$ denotes the number of failures in n_i independent Bernoulli trials. This set of data is used to compute the values of $\hat{q}_i$, $\hat{\alpha}_i$, $\hat{q}_{m,v(\alpha)}$ and $\hat{R}_{S,L(\alpha)}$ for each of the three interval estimation methods. Thus one realization of the lower bound estimate $\hat{R}_{L(\alpha),1}$ for the random variable $\hat{R}_{S,L(\alpha)}$ is obtained for each set of $\sum_{i=0}^{k} n_i$ uniform random numbers generated. Each realization described was replicated 1000 times in order to generate a simulated empirical population of this random variable $\hat{R}_{S,L(\alpha)}$. The vector of replications thus obtained was sorted from smallest to largest to obtain $\hat{R}_{L(\alpha),(1)}, \dots, \hat{R}_{L(\alpha),(1000)}$. Then the 100(1-$\alpha$)% approximate lower confidence bound on the true system reliability R_S is the 1000(1-α)th element of the sorted vector of replications; namely, $\hat{R}_{L(\alpha),(1000(1-\alpha))}$.

The simulated true confidence level is calculated by finding the element of the vector of replications which is closest to R_S and observing its index number j. Then the simulated "true" confidence level is $\frac{j}{1000} \times 100$. If there is a sequence of indices with the same value closest to R_S, we chose the smallest index j. This may yield an arbitrarily small value for our recorded "true" confidence level.

In addition to the external subroutines programmed by the author and to the ones mentioned above, the following subroutines were used:

- IMSL subroutine MDCHI was used to calculate the value of a chi-square random variable given the quantile and the degrees of freedom.
- NON-IMSL subroutine SHSORT was used to sort the arrays of system reliability estimates in ascending order in order to obtain the appropriate order statistic.
- IMSL subroutine USMNMX was used to extract the minimum and maximum values of the vector of mission tests for each component and of the vector of input component unreliabilities.

Each set of input parameters defines a simulation run, or a case. The following tables specify the sets of parameters used for each case that was simulated.

Table 1. INPUT NUMBER OF MISSION TRIALS N_i AND UNRELIABILITIES Q_i FOR A SERIES SYSTEM. CASES 1-3.

Component no.	n_i by Case Number			Input q_i (cases 1-3)
	1	2	3	
1	100	50	30	0.0200
2	30	30	25	0.0100
3	15	10	20	0.0050
4	10	10	10	0.0050
5	5	5	5	0.0050

Table 2. INPUT NUMBER OF MISSION TRIALS N_i AND UNRELIABILITIES Q_i FOR A SERIES SYSTEM. CASES 4-6.

Component no.	n_i by Case Number			Input q_i (cases 4-6)
	4	5	6	
1	100	50	30	0.0100
2	30	30	25	0.0050
3	15	10	20	0.0025
4	10	10	10	0.0025
5	5	5	5	0.0025

Table 3. INPUT NUMBER OF MISSION TRIALS N_i AND UNRELIABILITIES Q_i FOR A SERIES SYSTEM. CASES 7-9.

Component no.	n_i by Case Number			Input q_i (cases 7-9)
	7	8	9	
1	100	50	30	0.0200
2	30	30	25	0.0100
3	15	15	20	0.0050
4	10	5	15	0.0050
5	10	5	10	0.0050
6	10	5	5	0.0050
7	10	5	5	0.0050
8	10	5	5	0.0050
9	10	5	5	0.0050
10	10	5	5	0.0050

Table 4. INPUT NUMBER OF MISSION TRIALS N_i AND UNRELIABILITIES Q_i FOR A SERIES SYSTEM. CASES 10-12

Component no.	n_i by Case Number			Input q_i (cases 10-12)
	10	11	12	
1	100	50	30	0.0100
2	30	30	25	0.0050
3	15	15	20	0.0025
4	10	5	15	0.0025
5	10	5	10	0.0025
6	10	5	5	0.0025
7	10	5	5	0.0025
8	10	5	5	0.0025
9	10	5	5	0.0025
10	10	5	5	0.0025

Table 5. INPUT NUMBER OF MISSION TRIALS N_i AND UNRELIABILITIES Q_i FOR A WHEATSTONE BRIDGE. CASES 13-15.

Component no.	n_i by Case Number			Input q_i (cases 13-15)
	13	14	15	
1	100	50	30	0.1000
2	30	30	25	0.0700
3	15	10	20	0.0700
4	10	10	10	0.0700
5	5	5	5	0.0700

Table 6. **INPUT NUMBER OF MISSION TRIALS N_i AND UNRELIABILITIES Q_i FOR A WHEATSTONE BRIDGE. CASES 16-18.**

Component no.	**n_i by Case Number**			**Input q_i (cases 16-18)**
	16	**17**	**18**	
1	100	50	30	0.0500
2	30	30	25	0.0350
3	15	10	20	0.0350
4	10	10	10	0.0350
5	5	5	5	0.0350

The values of the scaling factor K used in Procedures 2 and 3 are as follows:

Table 7. **VALUES OF SCALING FACTOR K.**

Case Number	K
1-12	4
13-18	1.43

These values are computed from the assigned input parameters for these cases; e.g., $4 = \frac{.01}{.0025}$ from table 4 and $1.43 \simeq \frac{.05}{.035}$ from table 6.

IV. SIMULATION RESULTS

The results of the simulation runs for the twelve series system cases are presented in Tables 8 through 19. The results of the simulation runs for the six bridge system cases are presented in Tables 20 and 21. The accuracy of a particular interval estimation procedure for each case can be assessed from these tables. If the interval estimation procedure is exact, the two values R_S and $\hat{R}_{L(\alpha),1000(1-\alpha)}$ would be equal and the True Confidence Level value would be 100(1-α). For example, consider Table 8, case 2 and $\alpha = .20$. True system reliability, R_S is 0.95572 and $\hat{R}_{L(\alpha),1000(1-\alpha)} = 0.96398$. Thus the simulations indicate that the 80th percentile point of the distribution of $\hat{R}_{L(\alpha)}$ for Procedure 2 for case 2 is 0.964 instead of 0.956. This error reflects the inaccuracy of the approximate 80% lower confidence limit $\hat{R}_{S,L(.20)}$ as defined under Procedure 2. The table also indicates that $R_S \equiv 0.95572$ is approximately the 77.4 percentile point of the distribution of $\hat{R}_{S,L(.20)}$ instead of the 80th percentile point.

The number of components in the system that had zero failures in each of the 1000 replications was recorded. The average of the 1000 values is displayed in the last column of the table.

A. SERIES SYSTEM

Table 8. SUMMARY OF RESULTS FOR A SERIES SYSTEM-PROCEDURE 2, CASES 1-3

No. of Compnts	Case no.	Significance Level	R_S	$\hat{R}_{L(\alpha),1000(1-\alpha)}$	True Confidence Level	Avg. no. of comp. with no failure
5	1	$\alpha = .20$	0.95572	0.93563	93.20	3.71
		$\alpha = .05$	0.95572	0.93981	100.00	3.71
	2	$\alpha = .20$	0.95572	0.96398	77.40	3.98
		$\alpha = .05$	0.95572	0.93366	100.00	3.98
	3	$\alpha = .20$	0.95572	0.96633	65.40	4.16
		$\alpha = .05$	0.95572	0.93793	100.00	4.16

increase.

Table 9. SUMMARY OF RESULTS FOR A SERIES SYSTEM-PROCEDURE 2, CASES 4-6

No. of Compnts	Case no.	Signif-icance Level	R_S	$\hat{R}_{L(\alpha), 1000(1-\alpha)}$	True Con-fidence Level	Avg. no. of comp. with no failure
5	4	$\alpha = .20$	0.97768	0.96737	100.00	4.15
		$\alpha = .05$	0.97768	0.93981	100.00	4.15
	5	$\alpha = .20$	0.97768	0.96398	100.00	4.40
		$\alpha = .05$	0.97768	0.93366	100.00	4.40
	6	$\alpha = .20$	0.97768	0.96633	100.00	4.55
		$\alpha = .05$	0.97768	0.93793	100.00	4.55

Table 10. SUMMARY OF RESULTS FOR A SERIES SYSTEM-PROCEDURE 2, CASES 7-9

No. of Compnts	Case no.	Signif-icance Level	R_S	$\hat{R}_{L(\alpha), 1000(1-\alpha)}$	True Con-fidence Level	Avg. no. of comp. with no failure
10	7	$\alpha = .20$	0.93206	0.93884	77.30	8.49
		$\alpha = .05$	0.93206	0.94091	94.30	8.49
	8	$\alpha = .20$	0.93206	0.88770	81.00	8.86
		$\alpha = .05$	0.93206	0.89162	100.00	8.86
	9	$\alpha = .20$	0.93206	0.94925	70.60	9.01
		$\alpha = .05$	0.93206	0.90729	100.00	9.01

increase.

Table 11. SUMMARY OF RESULTS FOR A SERIES SYSTEM-PROCEDURE 2, CASES 10-12

No. of Compnts	Case no.	Significance Level	R_S	$\hat{R}_{L(\alpha),\ 1000(1-\alpha)}$	True Confidence Level	Avg. no. of comp. with no failure
10	10	$\alpha = .20$	0.96552	0.96788	75.30	9.03
		$\alpha = .05$	0.96552	0.94091	100.00	9.03
	11	$\alpha = .20$	0.96552	0.94046	100.00	9.36
		$\alpha = .05$	0.96552	0.89162	100.00	9.36
	12	$\alpha = .20$	0.96552	0.94925	100.00	9.48
		$\alpha = .05$	0.96552	0.90729	100.00	9.48

Table 12. SUMMARY OF RESULTS FOR A SERIES SYSTEM-PROCEDURE 1, CASES 1-3

No. of Compnts	Case no.	Significance Level	R_S	$\hat{R}_{L(\alpha),\ 1000(1-\alpha)}$	True Confidence Level	Avg. no. of comp. with no failure
5	1	$\alpha = .20$	0.95572	0.95721	65.20	3.70
		$\alpha = .05$	0.95572	0.95254	100.00	3.70
	2	$\alpha = .20$	0.95572	0.94011	100.00	3.97
		$\alpha = .05$	0.95572	0.90508	100.00	3.97
	3	$\alpha = .20$	0.95572	0.90018	100.00	4.14
		$\alpha = .05$	0.95572	0.84181	100.00	4.14

increase.

Table 13. SUMMARY OF RESULTS FOR A SERIES SYSTEM-PROCEDURE 1, CASES 4-6

No. of Compnts	Case no.	Signif- icance Level	R_S	$\hat{R}_{L(\alpha), 1000(1-\alpha)}$	True Con- fidence Level	Avg. no. of comp. with no failure
5	4	$\alpha = .20$	0.97768	0.97005	100.00	4.14
		$\alpha = .05$	0.97768	0.95254	100.00	4.14
	5	$\alpha = .20$	0.97768	0.94011	100.00	4.38
		$\alpha = .05$	0.97768	0.90508	100.00	4.38
	6	$\alpha = .20$	0.97768	0.88021	100.00	4.53
		$\alpha = .05$	0.97768	0.84181	100.00	4.53

Table 14. SUMMARY OF RESULTS FOR A SERIES SYSTEM-PROCEDURE 1, CASES 7-9

No. of Compnts	Case no.	Signif- icance Level	R_S	$\hat{R}_{L(\alpha), 1000(1-\alpha)}$	True Con- fidence Level	Avg. no. of comp. with no failure
10	7	$\alpha = .20$	0.93206	0.95721	59.80	8.48
		$\alpha = .05$	0.93206	0.95254	74.40	8.48
	8	$\alpha = .20$	0.93206	0.94011	75.90	8.85
		$\alpha = .05$	0.93206	0.90508	100.00	8.85
	9	$\alpha = .20$	0.93206	0.88021	100.00	8.99
		$\alpha = .05$	0.93206	0.84181	100.00	8.99

increase.

Table 15. SUMMARY OF RESULTS FOR A SERIES SYSTEM-PROCEDURE 1, CASES 10-12

No. of Compnts	Case no.	Significance Level	R_S	$\hat{R}_{L(\alpha),\,1000(1-\alpha)}$	True Confidence Level	Avg. no. of comp. with no failure
10	10	$\alpha = .20$	0.96552	0.97005	74.40	9.02
		$\alpha = .05$	0.96552	0.95254	100.00	9.02
	11	$\alpha = .20$	0.96552	0.94011	100.00	9.35
		$\alpha = .05$	0.96552	0.90508	100.00	9.35
	12	$\alpha = .20$	0.96552	0.88021	100.00	9.44
		$\alpha = .05$	0.96552	0.84181	100.00	9.44

Table 16. SUMMARY OF RESULTS FOR A SERIES SYSTEM-PROCEDURE 3, CASES 1-3

No. of Compnts	Case no.	Significance Level	R_S	$\hat{R}_{L(\alpha),\,1000(1-\alpha)}$	True Confidence Level	Avg. no. of comp. with no failure
5	1	$\alpha = .20$	0.95572	0.93563	93.20	3.70
		$\alpha = .05$	0.95572	0.93981	100.00	3.70
	2	$\alpha = .20$	0.95572	0.96398	77.30	3.97
		$\alpha = .05$	0.95572	0.93366	100.00	3.97
	3	$\alpha = .20$	0.95572	0.96633	65.40	4.14
		$\alpha = .05$	0.95572	0.93793	100.00	4.14

increase.

Table 17. SUMMARY OF RESULTS FOR A SERIES SYSTEM-PROCEDURE 3, CASES 4-6

No. of Compnts	Case no.	Significance Level	R_S	$\hat{R}_{L(\alpha), 1000(1-\alpha)}$	True Confidence Level	Avg. no. of comp. with no failure
5	4	$\alpha = .20$	0.97768	0.96737	100.00	4.14
		$\alpha = .05$	0.97768	0.93981	100.00	4.14
	5	$\alpha = .20$	0.97768	0.96398	100.00	4.38
		$\alpha = .05$	0.97768	0.93366	100.00	4.38
	6	$\alpha = .20$	0.97768	0.96633	100.00	4.53
		$\alpha = .05$	0.97768	0.93793	100.00	4.53

Table 18. SUMMARY OF RESULTS FOR A SERIES SYSTEM-PROCEDURE 3, CASES 7-9

No. of Compnts	Case no.	Significance Level	R_S	$\hat{R}_{L(\alpha), 1000(1-\alpha)}$	True Confidence Level	Avg. no. of comp. with no failure
10	7	$\alpha = .20$	0.93206	0.93884	77.30	8.48
		$\alpha = .05$	0.93206	0.94091	94.30	8.48
	8	$\alpha = .20$	0.93206	0.88770	80.80	8.85
		$\alpha = .05$	0.93206	0.89162	100.00	8.85
	9	$\alpha = .20$	0.93206	0.94925	70.40	8.99
		$\alpha = .05$	0.93206	0.90729	100.00	8.99

increase.

Table 19. SUMMARY OF RESULTS FOR A SERIES SYSTEM-PROCEDURE 3, CASES 10-12

No. of Compnts	Case no.	Significance Level	R_S	$\hat{R}_{L(\alpha),1000(1-\alpha)}$	True Confidence Level	Avg. no. of comp. with no failure
10	10	$\alpha = .20$	0.96552	0.96788	75.30	9.02
		$\alpha = .05$	0.96552	0.94091	100.00	9.02
	11	$\alpha = .20$	0.96552	0.94046	100.00	9.35
		$\alpha = .05$	0.96552	0.89162	100.00	9.35
	12	$\alpha = .20$	0.96552	0.94925	100.00	9.44
		$\alpha = .05$	0.96552	0.90729	100.00	9.44

B. BRIDGE SYSTEM

Table 20. SUMMARY OF RESULTS FOR A BRIDGE SYSTEM-PROCEDURE 1, CASES 13-15

No. of Compnts	Case no.	Significance Level	R_S	$\hat{R}_{L(\alpha),1000(1-\alpha)}$	True Confidence Level	Avg. no. of comp. with no failure
5	13	$\alpha = .20$	0.98742	0.99372	56.6	1.64
		$\alpha = .05$	0.98742	1.00000	71.50	1.64
	14	$\alpha = .20$	0.98742	0.99332	59.30	1.78
		$\alpha = .05$	0.98742	1.00000	77.20	1.78
	15	$\alpha = .20$	0.98742	0.99333	59.60	1.57
		$\alpha = .05$	0.98742	1.00000	80.10	1.57

increase.

Table 21. SUMMARY OF RESULTS FOR A BRIDGE SYSTEM-PROCEDURE 1, CASES 16-18

No. of Compnts	Case no.	Signif-icance Level	R_S	$\hat{R}_{L(\alpha),\,1000(1-\alpha)}$	True Con-fidence Level	Avg. no. of comp. with no failure
5	16	$\alpha = .20$	0.94816	0.96354	62.60	0.77
		$\alpha = .05$	0.94816	0.97426	77.70	0.77
	17	$\alpha = .20$	0.94816	0.96122	64.10	0.94
		$\alpha = .05$	0.94816	0.96740	82.80	0.94
	18	$\alpha = .20$	0.94816	0.95858	69.70	0.73
		$\alpha = .05$	0.94816	0.97125	86.30	0.73

V. SUMMARY AND CONCLUSIONS

The results in Tables 8 through 19 in section IV are quite mixed with respect to the accuracy across all twelve cases. However, these results definitely indicate that any one of the three interval estimation procedures offers excellent potential for a reasonably accurate lower confidence interval estimation method for system reliability. The accuracy of approximate discrete interval estimation methods for system reliability in general depend on the amount of testing relative to the expected number of failures, $\sum_{i=1}^{k} n_i q_i$. Woods and Borsting [Ref. 7] for example, show their lower confidence limit procedure to be quite accurate for cases for which the expected number of failures is greater than 5. The expected number of failures for some of the cases simulated in this thesis are as follows:

Table 22. EXPECTED NUMBER OF FAILURES

Case No.:	1	4	7	10	13	14	16
$\sum_{i=0}^{k} n_i q_i$	2.45	1.225	2.725	1.362	14.2	8.85	6.8

Consequently, any cases other than those numbered 13, 14, and 16 should not be expected to be highly accurate if these three procedures have features similar to those of the Woods-Borsting procedure [Ref. 7 : p.1]. Actually, in light of the small values of $\sum_{i=1}^{k} n_i q_i$ for cases 1, 4, 7, and 10, it is somewhat surprising that the procedures are as accurate as they are at the 80% level of confidence. The results for cases 13, 14, and 16 are quite good at the 80% level of confidence for the bridge system. Some of the parameter sets for the cases run were designed to detect instances where the methods might not be accurate.

The three procedures analyzed in this thesis should be modified and combined to establish one general procedure. The corresponding range of parameter sets for which the procedure is accurate should be established. This range will likely be a function of $\sum_{i=1}^{k} n_i q_i$. This kind of a procedure could be useful in many practical settings. Moreover, it could be combined with the method developed in the thesis by Lee [Ref. 10: pp.3-23]. Additional analysis, and simulation need to be performed to establish such a procedure.

APPENDIX A. INTERVAL ESTIMATION PROCEDURE 1

```
      PROGRAM ZFYSCN
**************************************************************************
*                                                                        *
*      TITLE: BINOMIAL INTERVAL ESTIMATION PROCEDURE                     *
*             ZERO FAILURES ALLOWED; NO SCALING                          *
*     AUTHOR: E. F. BELLINI, LT, USN                                     *
*       DATE: NOV 89                                                     *
*                                                                        *
*     THIS PROGRAM COMPUTES THE TRUE CONFIDENCE LEVEL FOR THE ESTIMATE   *
*     RELIABILITY OF A SERIES AND BRIDGE SYSTEM GIVEN THE RELIABILITY    *
*     OF THEIR COMPONENTS                                                *
*                                                                        *
*     IN ITS PRESENT CONFIGURATION THIS PROGRAM IS SET UP TO RUN 12      *
*     TIMES EACH TIME PRODUCING 1000 REPLICATIONS USING A DIFFERENT      *
*     SET OF INPUT DATA.  RUN THE PROGRAM FROM CMS BY TYPING 'B1 EXEC'.  *
*     THE REXX EXEC PROGRAM                                              *
*     'B1' CALLS THE INPUT FILES TO BE READ AND NAMES THE 12 OUTPUT      *
*     FILES RESULTING FROM THE 12 CONSECUTIVE RUNS.  BY EDITING THE      *
*     INDEX COUNTERS I, J, K OF THE 'B1' EXEC ONE CAN RUN ANY USER-      *
*     SPECIFIC RUN FROM JUST ONE RUN TO ALL 12.                          *
*                                                                        *
*     VARIABLES USED                                                     *
*                                                                        *
*          AHATI  : WEIGHT ESTIMATES FOR EACH COMPONENT                  *
*          AI     : INPUT WEIGHTS FOR EACH COMPONENT                     *
*          ALFA   : LEVELS OF SIGNIFICANCE                               *
*          BIGF   : TOTAL NO. OF FAILURES FOR EACH REPLICATION           *
*          CHISQ  : CHI-SQUARE RANDOM VARIABLE VALUE                     *
*          C1C15  : FORMAT LABEL                                         *
*          DEGFR  : DEGREES OF FREEDOM                                   *
*          DELBRG : DIFFERENCE FOR BRIDGE SYSTEM                         *
*          DELSTR : DIFFERENCE FOR SERIES SYSTEM- CLOSED FORM            *
*          DELTAR : DIFFERENCE FOR SERIES SYSTEM                         *
*          DIFF   : DIFFERENCE (TRUE REL. - ESTIMATED REL.)              *
*          EPS    : SMALL QUANTITY(CONSTANT)                             *
*          ERROR  : PARAMETER FOR IMSL ROUTINE                           *
*          FAILS  : COUNTS NO. OF REPLICATIONS WITH AT LST. 1 FAILURE    *
*          FI     : NO. OF FAILURES FOR EACH COMPONENT(ALL MISSION TST)*
*          FLAG   : 1 IF ALL COMP. HAVE SAME NO. OF MISSION TESTS        *
*          INC    : INCREMENT STEP SIZE FOR ROUTINE USMNMX               *
*          KEY1   : ARRAY OF INDECES FOR ROUTINE SHSORT                  *
*          KEY2   : ARRAY OF INDECES FOR ROUTINE SHSORT                  *
*          KEY3   : ARRAY OF INDECES FOR ROUTINE SHSORT                  *
*          KEY4   : ARRAY OF INDECES FOR ROUTINE SHSORT                  *
*          KK     : ARRAY SIZING PARAMETER FOR THE MAX NO OF COMPONENTS*
*          LOOP   : COUNTS NO. OF REPLICATION PERFORMED                  *
*          MAXALF : MAX NO. OF SIGNIFICANT LEVELS DESIRED(ARRAY SIZING)*
*          MAXREP : MAX NO. OF REPLICATIONS                              *
*          MAXRUN : MAX NO. OF PROGRAM ITERATIONS ALLOWED                *
*          MSTRQ  : MASTER UNRELIABILITY(USED WITH AI'S TO CALC. QI'S) *
*          MULT   : MULTIPLIER FOR RANDOM NO. GENERATOR SRN:             *
```

```
*          N       : NO. OF MISSION TEST FOR EACH COMPONENT              *
*          NIMAX   : MAX NO. OF MISSION TESTS                            *
*          NIMIM   : MIN NO. OF MISSION TESTS                            *
*          NINDX   : INDEX NO. OF MAX NO. OF MISSION TESTS               *
*          NIREAL  : NO. OF MISSION TESTS TRANSFORMED TO REAL            *
*          NMAX    : MAX NO. OF MISSION TESTS FOR OUTPUT CONTROL         *
*          NPRNT   : FLAG FOR DETAILED REPORT OUTPUT                     *
*          PRNT    : SAME AS ABOVE(PARAMETER)                            *
*          QHATI   : UNRELIABILITY ESTIMATES FOR EACH COMPONENT          *
*          QHTMAX  : LARGEST QHATI                                       *
*          QHTUPR  : UPPER LIMIT ON SYSTEM UNRELIABILITY                 *
*          QI      : INPUT UNRELIABILIY FOR EACH COMPONENT               *
*          QINDX   : INDEX                                               *
*          QUANTL  : QUANTILE                                            *
*          REPSHD  : REPLICATIONS HEADING FORMAT NUMBER                  *
*          RHTSTR  : SERIES SYSTEM RELIABILITY ESTIMATE(CLOSED FORM)     *
*          RS      : TRUE SERIES SYSTEM RELIABILITY                      *
*          RSBRDG  : TRUE BRIDGE SYSTEM RELIABILITY                      *
*          RSHAT   : SERIES SYSTEM RELIABILITY ESTIMATE                  *
*          RSHTBR  : BRIDGE SYSTEM RELIABILITY ESTIMATE                  *
*          SEED    : PARAMETER                                           *
*          SELCTA  : SIGNIFICANCE LEVEL SELECTION                        *
*          SELCTB  : SIGNIFICANCE LEVEL SELECTION                        *
*          SORT    : PARAMETER FOR ROUTINE SRND                          *
*          SUMNAI  : SUM OF THE PRODUCT OF NI'S AND AI'S                 *
*          TEMP    : TEMPORARY ARRAY                                     *
*          TOTREP  : TOTAL NUMBER OF PROGRAM ITERATIONS                  *
*          TRANBR  : TEMPORARY ARRAY                                     *
*          TRANSQ  : TEMPORARY ARRAY                                     *
*          TRANSR  : TEMPORARY ARRAY                                     *
*          TRIALS  : BERNOULLI TRIALS ARRAY (2-DIM)                      *
*          TRNSTR  : TEMPORARY ARRAY                                     *
*          TRUQNT  : TRUE QUANTILE                                       *
*          UNIRV   : UNIFORM RANDOM DEVIATES (2-DIM)                     *
*          ZFAILS  : TOTAL NUMBER OF REPLICATIONS WITH ZERO FAILURES     *
*          ZFPREP  : NO. OF COMPNTS. WITH ZERO FAILURES PER REPLICATION  *
*                                                                        *
**************************************************************************

      PARAMETER (KK=10,MAXALF=2,NPRNT=0)
      PARAMETER (MAXREP=1000, MAXRUN=2000, EPS=.000001)
      REAL*4 UNIRV(15,1000),TEMP(1000),QI(KK),AI(KK),AHATI(KK)
      REAL*4 QHATI(KK), NMAX, NNMAX, QHTMAX, CHISQR(5,5), ALFA(MAXALF)
      REAL*4 DF(5),AALFA(5),SUMNAI,RSHAT(MAXALF,MAXREP),RS
      REAL*4 KEY1(MAXREP),KEY2(MAXREP),KEY3(MAXREP),TRNSTR(MAXREP)
      REAL*4 DEGFR(MAXREP), QHTUPR(MAXALF,MAXREP),CHISQ(MAXALF,MAXREP)
      REAL*4 QUPA1(MAXREP), QUPA2(MAXREP),RHTSTR(MAXALF,MAXREP)
      REAL*4 DELTAR(MAXALF), TRANSQ(MAXREP),TRANSR(MAXREP),DIFF(MAXREP)
      REAL*4 DELSTR(MAXALF),NIMIN,NIMAX,NIREAL(KK)
      REAL*4 RSHTBR(MAXALF,MAXREP),DELBRG(MAXALF),KEY4(MAXREP)
      REAL*4 TRANBR(MAXREP), RSBRDG ,MSTRQ
      REAL*4 ZFPREP

      INTEGER SEED, MULT, SORT, TRIALS(15,1000), BIGF, FI(KK), N(KK)
      INTEGER NINDX, QINDX, ERROR, REPS, SELCTA, SELCTK, TOTREP
      INTEGER C1C15, REPSHD, SELCTB, ALF, FLAG, LOOP,PRNT
```

```
      INTEGER QUANTL(MAXALF), TRUQNT(MAXALF),ZFAILS,FAILS,INC

      DATA SEED/123457/, MULT/1/, INC/1/
      DATA AALFA/.01,.05,.9,.95,.99/, DF/1,5,10,30,40/
      DATA ALFA/.20,.050/
      DATA SORT/0/

      ASSIGN 8 TO C1C15
      ASSIGN 9 TO REPSHD

*     CALL COMPRS
      PRNT = NPRNT

      DO 12 I=1,KK
           AI(I) = 9999.
           N(I) = 99999999
   12 CONTINUE

      READ(03,*)K,MSTRQ

      DO 11 I=1,K
      READ(03,*) AI(I),N(I)
   11 CONTINUE


      IF(K.NE.5) THEN
           WRITE(1,'(''WARNING: BRIDGE STRUCTURE '',
     +''ONLY USES THE FIRST 5 COMPONENTS'')')
      ELSE
      END IF

***// INITIALIZE THE QHTUPR ARRAY OF UNRELIABILITY REPLICATIONS,   //***
*     RSHAT ARRAY OF ESTIMATE SYSTEM RELIABILITY REPLICATIONS          *
*     AND RHTSTR ARRAY OF EST. SYST. REL. FOR A SERIES SYST WHEN       *
***// ALL THE COMPONENT MISSION TESTS ARE EQUAL IN NUMBER          //***

      DO 172 ALF=1,MAXALF
           DO 173 REPS=1,MAXREP
                QHTUPR(ALF,REPS) = 0.
                RSHAT(ALF,REPS) = 0.
                RHTSTR(ALF,REPS) = 0.
                RSHTBR(ALF,REPS) = 0.
 173       CONTINUE
 172  CONTINUE

***// SET FLAG TO 1 IF ALL COMPONENTS HAVE SAME NO. OF MISSION TESTS****

      FLAG=1
      DO 50 I=1,K -1
           IF((N(I) - N(I+1)).NE.0) THEN
                FLAG=0
           ELSE
           END IF
   50 CONTINUE
      PRINT *, 'FLAG IS:', FLAG
***// MAIN PROGRAM OUTER LOOP START(EVERY LOOP IS ONE REPLICATION)//***
```

```
      ZFPREP = 0.
      ZFAILS = 0
      FAILS = 0
      TOTREP = 0
      LOOP = 0
 10   IF(LOOP.LT.MAXREP) THEN
           LOOP = LOOP + 1
           IF(TOTREP.LT.MAXRUN) THEN
                TOTREP = TOTREP + 1
      SELCTA = 1
      SELCTB = 2
***// FILL ARRAY KEY(REPS) WITH INTEGERS 1 TO K TO BE USED AS OUTPUT
***// OF THE SUBROUTINE SHSORT
      DO 95 REPS=1, MAXREP
           KEY1(REPS) = REPS
           KEY2(REPS) = REPS
           KEY3(REPS) = REPS
           KEY4(REPS) = REPS
 95   CONTINUE

***// CALCULATE NMAX NOT TO PRINT LONGER THAN THE MAX SAMPLE SIZE

***// CALCULATE THE MAXIMUM NO. OF TRIALS AND ITS INDEX NO. //***

      CALL IMAX(N,K,NMAX,NINDX)


***// CALCULATE THE QI'S FROM THE GIVEN MASTER Q AND THE AI'S

      DO 115 I=1, K
           QI(I) = MSTRQ * AI(I)
 115  CONTINUE

      DO 120 I=1,15
           DO 125 J=1,500
                UNIRV(I,J) = 999.
                TRIALS(I,J) = 99999
 125       CONTINUE
 120  CONTINUE

***// DRAW UNIFORM (0,1) RV'S AND CONVERT TO BERNOULLI TRIALS //***

      DO 130 I=1, K
           CALL SRND(SEED, TEMP, N(I), MULT, SORT)
           DO 135 J=1, N(I)
                UNIRV(I,J) = TEMP(J)
                IF (UNIRV(I,J).LE. 1 - QI(I)) THEN
                     TRIALS(I,J) = 0
                ELSE
                     TRIALS(I,J) = 1
                END IF
 135  CONTINUE
 130  CONTINUE
```

```
***// CALCULATE THE NO. OF FAILURES FOR EACH COMPONENT  //***

      DO 150 I=1, K
          FI(I) = 0
 150  CONTINUE

***// CALCULATE THE F SUB I'S AND THE GRAND TOTAL NO. OF FAILURES
      BIGF = 0
      DO 155 I=1, K
          DO 160 J=1, N(I)
              FI(I) = FI(I) + TRIALS(I,J)
 160        CONTINUE
          IF(FI(I).EQ.0) THEN
              ZFPREP = ZFPREP + 1
          ELSE
          END IF
***// CALCULATE THE QHAT SUB I'S: F SUB I'S DIVIDED BY N SUB I'S
              QHATI(I) = REAL(FI(I)) / N(I)
          BIGF = BIGF + FI(I)
 155  CONTINUE

***// CASE WHERE NO COMPONENTS HAVE ANY FAILURES                //***

      IF(BIGF.EQ.0) THEN
          ZFAILS = ZFAILS + 1
          DO 200 I=1, K
              NIREAL(I) = REAL(N(I))
 200      CONTINUE
          CALL USMNMX(NIREAL,K,INC,NIMIN,NIMAX)
          DO 205 ALF=1, MAXALF
              CALL MDCHI(1 - ALFA(ALF),2.,CHISQ(ALF,LOOP),ERROR)
              RSHAT(ALF,LOOP)= 1 - (CHISQ(ALF,LOOP) / REAL(2 * NIMIN))
          IF(FLAG.EQ.1) THEN
              RHTSTR(ALF,LOOP)=1-(CHISQ(ALF,LOOP) / REAL(2 * N(1)))
          ELSE
          END IF
 205      CONTINUE
      IF(PRNT.EQ.1) THEN
          WRITE(1,0007)
          WRITE(1,C1C15)
          WRITE(1,3334) QI
          WRITE(1,0001)
          WRITE(1,C1C15)
          DO 141 J=1,NMAX
              WRITE(1,1111) (UNIRV(I,J), I=1, K)
 141      CONTINUE
          WRITE(1,0002)
          WRITE(1,C1C15)
          DO 146 J=1,NMAX
              WRITE(1,2222) (TRIALS(I,J), I=1, K)
 146      CONTINUE
          WRITE(1,0003)
          WRITE(1,C1C15)
          WRITE(1,3333) FI
          WRITE(1,0005)
          WRITE(1,C1C15)
```

```
          WRITE(1,3335) N
          WRITE(1,0004)
          WRITE(1,C1C15)
          WRITE(1,3334) QHATI
      WRITE(1,'(/''THE MAXIMUM Q HAT SUB I IS:'', T40, F8.5)') QHTMAX
      WRITE(1,'(/''THE MAXI Q HAT SUB I IS ELMNT NO.:'',T40,I5)') QINDX
      WRITE(1,'(/''THE GRAND TOTAL NO. OF FAILURES IS:'',T40, I5)') BIGF
      ELSE
      ENDIF
               DEGFR(LOOP) = 2.
               GO TO 10
      ELSE
          FAILS = FAILS + 1
      END IF

***// FIND THE MAX OF THE INDIVIDUAL COMPONENT UNRELIABILITIES
      CALL RMAX(QHATI, K, QHTMAX, QINDX)

***// PRINT THE RESULT OF THE MAIN OPERATING ELEMENTS OF THE PROGRAM

      IF(PRNT.EQ.1) THEN
          WRITE(1,0007)
          WRITE(1,C1C15)
          WRITE(1,3334) QI
          WRITE(1,0001)
          WRITE(1,C1C15)
          DO 140 J=1,NMAX
               WRITE(1,1111) (UNIRV(I,J), I=1, K)
 140      CONTINUE
          WRITE(1,0002)
          WRITE(1,C1C15)
          DO 145 J=1,NMAX
               WRITE(1,2222) (TRIALS(I,J), I=1, K)
 145      CONTINUE
          WRITE(1,0003)
          WRITE(1,C1C15)
          WRITE(1,3333) FI
          WRITE(1,0005)
          WRITE(1,C1C15)
          WRITE(1,3335) N
          WRITE(1,0004)
          WRITE(1,C1C15)
          WRITE(1,3334) QHATI
      WRITE(1,'(/''THE MAXIMUM Q HAT SUB I IS:'', T40, F8.5)') QHTMAX
      WRITE(1,'(/''THE MAXI Q HAT SUB I IS ELMNT NO.:'',T40,I5)') QINDX
      WRITE(1,'(/''THE GRAND TOTAL NO. OF FAILURES IS:'',T40, I5)') BIGF
      ELSE
      ENDIF

***// CALCULATE THE AHAT SUB I'S (WEIGHT ESTIMATES)

      SUMNAI = 0.
      DO 165 I=1, K
          AHATI(I) = QHATI(I) / QHTMAX
          SUMNAI = SUMNAI + N(I) * AHATI(I)
 165  CONTINUE
```

```
      IF(PRNT.EQ.1) THEN
           WRITE(1,0006)
           WRITE(1,C1C15)
           WRITE(1,3334) AHATI
      ELSE
      END IF

***// CALCULATE 1 REPLICATION OF UPPR ALFA C.L. ON SYSTEM RELIABILITY

      DEGFR(LOOP) = 2 * (1 + BIGF)

      DO 170 ALF=1, MAXALF
           CALL MDCHI(1 - ALFA(ALF),DEGFR(LOOP),CHISQ(ALF,LOOP), ERROR)
           QHTUPR(ALF,LOOP) = CHISQ(ALF,LOOP) / (2 * SUMNAI)
           IF(FLAG.EQ.1) THEN
                RHTSTR(ALF,LOOP) = 1 -(CHISQ(ALF,LOOP) / REAL(2*N(1)))
           ELSE
           END IF
*    +     (ALF,LOOP), ALFA(ALF)
***// CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR COMPNTS. IN SERIES

           CALL RHTSRS(QHTUPR(ALF,LOOP), AHATI,K, RSHAT(ALF,LOOP))
*    +T40,F8.5)') RSHAT(ALF,LOOP)

***// CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR BRIDGE STRUCTURE ***

           CALL RHTBRG(QHTUPR(ALF,LOOP),AHATI,K,RSHTBR(ALF,LOOP))
 170  CONTINUE

***// THIS ELSE AND ENDIF ARE FOR THE TEST AGAINST MAXRUN   ***********
      ELSE
           WRITE(1,'('' '',/''PROGRAM EXCEEDED THE MAX NO. OF RUNS'',
     +'' ALLOWED OF:'',I6)') TOTREP
      GOTO 9999

      END IF
      GOTO 10
      END IF

***// SORT THE ARRAYS OF SYSTEM UNRELIABILITIES(1 FOR EACH CONF. LEVEL)

      DO 700 ALF=1, MAXALF
           DO 800 REPS=1, MAXREP
                TRANSQ(REPS) = QHTUPR(ALF,REPS)
                TRANSR(REPS) = RSHAT(ALF,REPS)
                TRNSTR(REPS) = RHTSTR(ALF,REPS)
                TRANBR(REPS) = RSHTBR(ALF,REPS)
 800       CONTINUE
           CALL SHSORT(TRANSQ,KEY1,MAXREP)
           CALL SHSORT(TRANSR,KEY2,MAXREP)
           CALL SHSORT(TRNSTR,KEY3,MAXREP)
           CALL SHSORT(TRANBR,KEY4,MAXREP)
           DO 900 REPS=1, MAXREP
                QHTUPR(ALF,REPS) = TRANSQ(REPS)
                RSHAT(ALF,REPS) = TRANSR(REPS)
                RHTSTR(ALF,REPS) = TRNSTR(REPS)
```

```
              RSHTBR(ALF,REPS) = TRANBR(REPS)
 900       CONTINUE
 700  CONTINUE


***// PRINT OUTPUT REPORT HEADINGS *************

      WRITE(1,6666)
      WRITE(1,6667) MAXREP
      WRITE(1,6668) K
      WRITE(1,6669)
      IF(K.EQ.5) THEN
           WRITE(1,6699)
      ELSE
      END IF
      WRITE(1,6670) MSTRQ
      WRITE(1,6671)
      WRITE(1,C1C15)
      WRITE(1,3334) AI
      WRITE(1,0007)
      WRITE(1,C1C15)
      WRITE(1,3334) QI
      WRITE(1,0005)
      WRITE(1,C1C15)
      WRITE(1,3335) N
      WRITE(1,6674)
      WRITE(2,6666)
      WRITE(2,6667) MAXREP
      WRITE(2,6668) K
      WRITE(2,6669)
      IF(K.EQ.5) THEN
           WRITE(1,6699)
      ELSE
      END IF
      WRITE(2,6670) MSTRQ
      WRITE(2,6671)
      WRITE(2,C1C15)
      WRITE(2,3334) AI
      WRITE(2,0007)
      WRITE(2,C1C15)
      WRITE(2,3334) QI
      WRITE(2,0005)
      WRITE(2,C1C15)
      WRITE(2,3335) N
      WRITE(2,6674)

      WRITE(2,'(''SORTED RSHAT 1  IS:'',/10(F8.5))')
     +(RSHAT(1,REPS), REPS=1, MAXREP)
      WRITE(2,'(''SORTED RSHAT 2  IS:'',/10(F8.5))')
     +(RSHAT(2,REPS), REPS=1, MAXREP)
      IF(FLAG.EQ.1) THEN
           WRITE(2,'(''SORTED RHTSTR 1  IS:'',/10(F8.5))')
     +(RHTSTR(1,REPS), REPS=1, MAXREP)
           WRITE(2,'(''SORTED RHTSTR 2  IS:'',/10(F8.5))')
     +(RHTSTR(2,REPS), REPS=1, MAXREP)
      ELSE
```

```
        END IF
        IF(K.EQ.5) THEN
             WRITE(2,'(''SORTED RSHTBR 1  IS:'',/10(F8.5))')
       +(RSHTBR(1,REPS), REPS=1, MAXREP)
             WRITE(2,'(''SORTED RSHTBR 2  IS:'',/10(F8.5))')
       +(RSHTBR(2,REPS), REPS=1, MAXREP)
        ELSE
        END IF

***// COMPUTE THE VALUE RS OF THE TRUE SYSTEM REL. FNCTN.(SERIES SYSTEM)
***// AND FOR THE 5-COMPONENT BRIDGE STRUCTURE
        CALL RSRS(QI,K,RS)
        WRITE(1,'('' '',////''THE TRUE SERIES SYSTEM '',
       +''RELIABILITY VALUE IS:'',T51,F8.5)') RS
        CALL RBRIDG(QI,K,RSBRDG)
        IF(K.EQ.5) THEN
        WRITE(1,'('' '',////''THE TRUE BRIDGE STRUCTURE '',
       +''RELIABILITY VALUE IS:'',T51,F8.5)') RSBRDG
        ELSE
        END IF
        WRITE(1,6675)

***// COMPUTE THE DIFFERENCE 'DELTAR' BTWN. RS AND RSHAT OF THE THEO
***// RETICAL QUANTILE GIVEN BY ALFA(MUST USE SORTED RSHAT ARRAY)

        IF(FLAG.EQ.1) THEN
        WRITE(1,5755)
        ELSE
        END IF
        DO 450 ALF=1, MAXALF
             QUANTL(ALF) = MAXREP * (1 - ALFA(ALF))
             DELTAR(ALF) = RS - RSHAT(ALF,QUANTL(ALF))
             DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))
             IF(FLAG.EQ.1) THEN
                  DELSTR(ALF) = RS - RHTSTR(ALF,QUANTL(ALF))
                  WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
                  WRITE(1,5656) RHTSTR(ALF,QUANTL(ALF))
                  WRITE(1,5657) DELSTR(ALF)
             ELSE
             END IF
             IF(K.EQ.5) THEN
                  DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))
                  WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
                  WRITE(1,5666) RSHTBR(ALF,QUANTL(ALF))
                  WRITE(1,5667) DELBRG(ALF)
             ELSE
             END IF
             WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
             WRITE(1,5556) RSHAT(ALF,QUANTL(ALF))
             WRITE(1,5557) DELTAR(ALF)
  450   CONTINUE
        PRINT *, 'QUANTL(1) IS:', QUANTL(1)
        PRINT *, 'QUANTL(2) IS:', QUANTL(2)

***// FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE //***
***//                  ********** RSHAT ********                   //***
```

```
      WRITE(1,6676)
      DO 400 ALF=1,MAXALF
      TRUQNT(ALF) = 0
           DO 500 REPS=1, MAXREP
                DIFF(REPS) = RS - RSHAT(ALF,REPS)
 500       CONTINUE
           DO 600 REPS=1, MAXREP
                IF(ABS(DIFF(REPS)).LE.EPS) THEN
                     TRUQNT(ALF) = REPS
                     WRITE(1,'('' '',/''TRUE CONFIDENCE LIMIT IS:'',
     +               F8.4)')
     +               (TRUQNT(ALF) / REAL(MAXREP)) * 100.
                     GO TO 620
                ELSEIF(DIFF(REPS).LT.0.) THEN
                     TRUQNT(ALF) = REPS
                     GO TO 610
                ELSE
                END IF
 600       CONTINUE
 610       IF(TRUQNT(ALF).EQ.0.) THEN
                WRITE(1,4443) ALFA(ALF)
                WRITE(1,'('' '',/''THE SMALLEST'',
     +          '' DIFFERENCE BETWEEN RS AND RSHAT IS:'',F10.5)') DIFF(
     +          MAXREP)
           ELSEIF(TRUQNT(ALF).EQ.1.) THEN
                WRITE(1,4442) ALFA(ALF)
                WRITE(1,'('' '',/''ALL RSHAT'',
     +           '' ARE GREATER THAN RS'')')
           ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))
     +     THEN
                WRITE(1,4444) ALFA(ALF),
     +          (TRUQNT(ALF) / REAL(MAXREP)) * 100.
                WRITE(1,4445) RSHAT(ALF,TRUQNT(ALF))
                WRITE(1,4446)
           ELSE
                WRITE(1,4444) ALFA(ALF),
     +          ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
                WRITE(1,4445) RSHAT(ALF,TRUQNT(ALF)-1)
                WRITE(1,4447)
 620       END IF
 400  CONTINUE


***// FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE //***
***//         ********* RSHTBR (BRIDGE) ********                   //***

      IF(K.EQ.5) THEN
      DO 401 ALF=1,MAXALF
      TRUQNT(ALF) = 0
           DO 501 REPS=1, MAXREP
                DIFF(REPS) = RSBRDG - RSHTBR(ALF,REPS)
 501       CONTINUE
           DO 601 REPS=1, MAXREP
                IF(ABS(DIFF(REPS)).LE.EPS) THEN
                     TRUQNT(ALF) = REPS
     -               WRITE(1,'('' '',/''TRUE CONFIDENCE LIMIT IS:'',
```

```
+                 F8.4)')
+                 (TRUQNT(ALF) / REAL(MAXREP)) * 100.
                  GO TO 621
             ELSEIF(DIFF(REPS).LT.0.) THEN
                  TRUQNT(ALF) = REPS
                  GO TO 611
             ELSE
             END IF
 601       CONTINUE
 611       IF(TRUQNT(ALF).EQ.0.) THEN
             WRITE(1,4443) ALFA(ALF)
             WRITE(1,'('' '',/''THE SMALLEST'',
+            '' DIFFERENCE BETWEEN RSBRDG AND RSHTBR IS:'',
+            F10.5)') DIFF(MAXREP)
           ELSEIF(TRUQNT(ALF).EQ.1.) THEN
             WRITE(1,4442) ALFA(ALF)
             WRITE(1,'('' '',/''ALL RSHTBR'',
+             '' ARE GREATER THAN RSBRDG'')')
           ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))
+          THEN
             WRITE(1,4444) ALFA(ALF),
+            (TRUQNT(ALF) / REAL(MAXREP)) * 100.
             WRITE(1,4449) RSHTBR(ALF,TRUQNT(ALF))
             WRITE(1,4446)
           ELSE
             WRITE(1,4444) ALFA(ALF),
+            ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
             WRITE(1,4449) RSHTBR(ALF,TRUQNT(ALF)-1)
             WRITE(1,4447)
 621       END IF
 401  CONTINUE
      ELSE
      END IF


***// FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE //***
***//                 ********* RHTSTR *********                   //***

      IF(FLAG.EQ.1) THEN
      DO 4400 ALF=1,MAXALF
      TRUQNT(ALF) = 0
           DO 5500 REPS=1, MAXREP
                DIFF(REPS) = RS - RHTSTR(ALF,REPS)
 5500      CONTINUE
           DO 6600 REPS=1, MAXREP
                IF(ABS(DIFF(REPS)).LE.EPS) THEN
                     TRUQNT(ALF) = REPS
                     WRITE(1,'('' '',/''TRUE CONFIDENCE LIMIT IS:'',
+                    F8.4)')
+                    (TRUQNT(ALF) / REAL(MAXREP)) * 100.
                     GO TO 6620
                ELSEIF(DIFF(REPS).LT.0.) THEN
                     TRUQNT(ALF) = REPS
                     GO TO 6610
                ELSE
                END IF
```

```
6600      CONTINUE
6610      IF(TRUQNT(ALF).EQ.0.) THEN
               WRITE(1,4443) ALFA(ALF)
               WRITE(1,'('' '',/''THE SMALLEST'',
     +         '' DIFFERENCE BETWEEN RS AND RHTSTR IS:'',
     +         F9.5)') DIFF(MAXREP)
          ELSEIF(TRUQNT(ALF).EQ.1.) THEN
               WRITE(1,4442) ALFA(ALF)
               WRITE(1,'('' '',/''ALL RHTSTR'',
     +         '' ARE GREATER THAN RS'')')
          ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))
     +    THEN
               WRITE(1,4444) ALFA(ALF),
     +         (TRUQNT(ALF) / REAL(MAXREP)) * 100.
               WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF))
               WRITE(1,4446)
          ELSE
               WRITE(1,4444) ALFA(ALF),
     +         ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
               WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF)-1)
               WRITE(1,4447)
6620      END IF
4400 CONTINUE
     ELSE
     END IF

***// PRINT THE ARRAYS PERTINENT TO THE OUPUT OF EACH REPLICATION *****

      IF(PRNT.EQ.1) THEN
      I = 1
 185  WRITE(1,REPSHD) ALFA(SELCTA), ALFA(SELCTA),
     +ALFA(SELCTB),ALFA(SELCTB),ALFA(SELCTA),ALFA(SELCTA),ALFA(SELCTB),
     +ALFA(SELCTB)
 175  IF(I.GE.(MAXREP + 1)) THEN
           GOTO 180
      ELSE
           IF( (I.EQ.71).OR.(I.EQ.211).OR.(I.EQ.351).OR.(I.EQ.491).OR.
     +     (I.EQ.631).OR.(I.EQ.771).OR.(I.EQ.911).OR.(I.EQ.1051) ) THEN
                I = I + 70
                WRITE(1,'(''+'')')
                GOTO 185
           ELSE

           WRITE(1,3336) I, INT(DEGFR(I)), CHISQ(1,I), QHTUPR(1,I),
     +     CHISQ(2,I), QHTUPR(2,I)
           END IF
           IF(I + 70.LE.MAXREP) THEN
                WRITE(1,3337) I+70,INT(DEGFR(I+70)),CHISQ(1,I+70),
     +          QHTUPR(1,I+70),CHISQ(2,I+70),QHTUPR(2,I+70)
           ELSE
           END IF
      I = I + 1
      GOTO 175
 180  END IF
      ELSE
      ENDIF
```

```
9999 WRITE(1,'(''THE TOTAL NO OF REPS WAS:'',I8)') TOTREP
     WRITE(1,'(''THE TOTAL NO OF EFFECTIVE REPS WAS:'',I8)') LOOP
     WRITE(1,'(''THE TOTAL NO OF NO FAILURE RUNS WAS:'',I8)') ZFAILS
     WRITE(1,'(''AVERAGE  NO. OF COMPONENTS PER REPLICATION WITH '',
    +''NO FAILURES:'',F5.2)') ZFPREP / MAXREP
     WRITE(1,'(''THE TOTAL NO OF RUNS WITH FAILURES WAS:'',I8)') FAILS
0008 FORMAT (/ 3X,'C 1',5X,'C 2',
    +5X,'C 3',5X,'C 4',5X,'C 5',5X,'C 6',5X,'C 7',5X,
    +'C 8',5X,'C 9',5X,'C 10',4X,'C 11',4X,
    +'C 12',4X,'C 13',4X,'C 14',4X,'C 15')
0009 FORMAT(/1X,'REP NO',2X,'DF',1X,'CHISQR(',F4.3,')',1X,
    +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'QHTUPR(',F4.3,')',
    +2X,'REP NO',2X,'DF',1X,'CHISQR(',F4.3,')',1X,
    +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'QHTUPR(',F4.3,')'/)
0001 FORMAT (///'UNIFORM RANDOM DEVIATES ARE:')
0002 FORMAT (///'BERNOULLI TRIALS ARE:')
0003 FORMAT (///'TOTAL NO. OF FAILURES FOR EACH COMPONENT:')
0004 FORMAT (///'ESTIMATED UNRELIABILITY FOR EACH COMPONENT:')
0005 FORMAT (///'TOTAL NUMBER OF MISSION TESTS:')
0006 FORMAT (///'ESTIMATED WEIGHTS FOR EACH COMPONENT:')
0007 FORMAT (///'Q I FOR EACH COMPONENT:')
1111 FORMAT (15F8.5)

2222 FORMAT (/1X,15(I4,4X))
3333 FORMAT (/1X,15(I4,4X))
3334 FORMAT (/15F8.5)
3335 FORMAT (/1X,15(I4,4X))
3336 FORMAT (T3,I4,T9,I3,T13,F11.5,T27,F8.5,T39,F11.5,T53,F8.5)
3337 FORMAT ('+',T67,I4,T73,I3,T77,F11.5,T91,F8.5,T103,F11.5,T117,F8.5)
4442 FORMAT (' ',///'THE RESULTING (1 - ',F4.3,') CONFIDENCE ',
    +'LIMIT IS:',T50,' 00.000 ')
4443 FORMAT (' ',///'THE RESULTING (1 - ',F4.3,') CONFIDENCE ',
    +'LIMIT IS:',T50,'100.0000')
4444 FORMAT (' ',///'THE RESULTING (1 - ',F4.3,') CONFIDENCE ',
    +'LIMIT IS:',T50,F8.4)
4445 FORMAT (' ',/'THE RSHAT VALUE CLOSEST TO RS IS: ',T51,F8.5)
4446 FORMAT (' ',/'(FIRST NEGATIVE DIFFERENCE)')
4447 FORMAT (' ',/'(ELEMENT PRECEEDING FIRST NEGATIVE DIFFERENCE)')
4448 FORMAT (' ',/'THE RHTSTR VALUE CLOSEST TO RS IS: ',T51,F8.5)
4449 FORMAT (' ',/'THE RSHTBR VALUE CLOSEST TO RSBRDG IS: ',T51,F8.5)
5555 FORMAT (' ',///'THE ',I4,'(1-',F4.3,') QUANTILE IS:',T49,F8.3)
5556 FORMAT (' ',/'THE VALUE OF RSHAT FOR THAT QUANTILE IS:',T51,F8.5)
5557 FORMAT (' ',/'THE DIFFERENCE(RS - RSHAT) IS:',T51,F8.5)
5656 FORMAT (' ',/'THE VALUE OF RHTSTR FOR THAT QUANTILE IS:',T51,F8.5)
5666 FORMAT (' ',/'THE VALUE OF RSHTBR FOR THAT QUANTILE IS:',T51,F8.5)
5657 FORMAT (' ',/'THE DIFFERENCE(RS - RHTSTR) IS:',T51,F8.5)
5667 FORMAT (' ',/'THE DIFFERENCE(RS - RSHTBR) IS:',T51,F8.5)
5755 FORMAT (' ',///'SINCE THE NO. OF MISSION TESTS IS THE SAME FOR',
    +' ALL COMPONENTS THE CLOSED FORM SERIES SYSTEM RELIABILITY ',
    +'''RHTSTR'' IS COMPUTED')
6666 FORMAT ('+','**************************************************',
    +'************ RUN INPUT SETTINGS ************************************',
    +'************************')
6667 FORMAT (' ',//'NUMBER OF REPLICATIONS:',T50,I4)
6668 FORMAT (' ',//'NUMBER OF COMPONENTS:',T50,I4)
6669 FORMAT (' ',//'SYSTEM RELIABILITY FUNCTION:',T50,'SERIES')
```

```
6699 FORMAT (' ',//'SYSTEM RELIABILITY FUNCTION:',T50,'BRIDGE')
6670 FORMAT (' ',//'MASTER UNRELIABILITY USED:',T50,F8.5)
6671 FORMAT (' ',//'INPUT WEIGHTS(A SUB I''S):')
6674 FORMAT ('+',///'**************************************************',
    +'**R U N  R E S U L T S*******************************************',
    +'***************')
6675 FORMAT ('+',///'***************************************************',
    +' ESTIMATE ERRORS ************************************************',
    +'***************')
6676 FORMAT ('+',///'**************************************************',
    +' TRUE CONFIDENCE LIMITS ******************************************',
    +'***************')
     END
```

APPENDIX B. INTERVAL ESTIMATING PROCEDURE 2

```
      PROGRAM ZFNSCY
**************************************************************************
*                                                                        *
*       TITLE: BINOMIAL INTERVAL ESTIMATION PROCEDURE                    *
*              ZERO FAILURES DISALLOWED; WITH SCALING                    *
*      AUTHOR: E. F. BELLINI, LT, USN                                    *
*        DATE: NOV 89                                                    *
*                                                                        *
*      THIS PROGRAM COMPUTES THE TRUE CONFIDENCE LEVEL FOR THE ESTIMATE *
*      RELIABILITY OF A SERIES AND BRIDGE SYSTEM GIVEN THE RELIABILITY  *
*      OF THEIR COMPONENTS                                               *
*                                                                        *
*      IN ITS PRESENT CONFIGURATION THIS PROGRAM IS SET UP TO RUN 12    *
*      TIMES EACH TIME PRODUCING 1000 REPLICATIONS USING A DIFFERENT    *
*      SET OF INPUT DATA.  RUN THE PROGRAM FROM CMS BY TYPING 'B1 EXEC'.*
*      THE REXX EXEC PROGRAM                                             *
*      'B1' CALLS THE INPUT FILES TO BE READ AND NAMES THE 12 OUTPUT    *
*      FILES RESULTING FROM THE 12 CONSECUTIVE RUNS.  BY EDITING THE    *
*      INDEX COUNTERS I, J, K OF THE 'B1' EXEC ONE CAN RUN ANY USER-    *
*      SPECIFIC RUN FROM JUST ONE RUN TO ALL 12.                        *
*                                                                        *
*      VARIABLES USED                                                    *
*                                                                        *
*           AHATI  : WEIGHT ESTIMATES FOR EACH COMPONENT                *
*           AI     : INPUT WEIGHTS FOR EACH COMPONENT                   *
*           ALFA   : LEVELS OF SIGNIFICANCE                             *
*           BIGF   : TOTAL NO. OF FAILURES FOR EACH REPLICATION         *
*           CHISQ  : CHI-SQUARE RANDOM VARIABLE VALUE                   *
*           COUNT1 : COUNTS THE NO. OF COMPONENTS WITH FAILURES         *
*           C1C15  : FORMAT LABEL                                       *
*           DEGFR  : DEGREES OF FREEDOM                                 *
*           DELBRG : DIFFERENCE FOR BRIDGE SYSTEM                       *
*           DELSTR : DIFFERENCE FOR SERIES SYSTEM- CLOSED FORM          *
*           DELTAR : DIFFERENCE FOR SERIES SYSTEM                       *
*           DIFF   : DIFFERENCE (TRUE REL. - ESTIMATED REL.)            *
*           EPS    : SMALL QUANTITY(CONSTANT)                           *
*           ERROR  : PARAMETER FOR IMSL ROUTINE                         *
*           FAILS  : COUNTS NO. OF REPLICATIONS WITH AT LST. 1 FAILURE  *
*           FI     : NO. OF FAILURES FOR EACH COMPONENT(ALL MISSION TST)*
*           FLAG   : 1 IF ALL COMP. HAVE SAME NO. OF MISSION TESTS      *
*           INC    : INCREMENT STEP SIZE FOR ROUTINE USMNMX             *
*           KEY1   : ARRAY OF INDECES FOR ROUTINE SHSORT                *
*           KEY2   : ARRAY OF INDECES FOR ROUTINE SHSORT                *
*           KEY3   : ARRAY OF INDECES FOR ROUTINE SHSORT                *
*           KEY4   : ARRAY OF INDECES FOR ROUTINE SHSORT                *
*           KK     : ARRAY SIZING PARAMETER FOR THE MAX NO OF COMPONENTS*
*           LOOP   : COUNTS NO. OF REPLICATION PERFORMED                *
*           MAXALF : MAX NO. OF SIGNIFICANT LEVELS DESIRED(ARRAY SIZING)*
*           MAXREP : MAX NO. OF REPLICATIONS                            *
*           MAXRUN : MAX NO. OF PROGRAM ITERATIONS ALLOWED              *
*           MSTRQ  : MASTER UNRELIABILITY(USED WITH AI'S TO CALC. QI'S) *
```

```
*          MULT   : MULTIPLIER FOR RANDOM NO. GENERATOR SRND                 *
*          N      : NO. OF MISSION TEST FOR EACH COMPONENT                   *
*          NIMAX  : MAX NO. OF MISSION TESTS                                 *
*          NIMIM  : MIN NO. OF MISSION TESTS                                 *
*          NINDX  : INDEX NO. OF MAX NO. OF MISSION TESTS                    *
*          NIREAL : NO. OF MISSION TESTS TRANSFORMED TO REAL                 *
*          NMAX   : MAX NO. OF MISSION TESTS FOR OUTPUT CONTROL              *
*          NPRNT  : FLAG FOR DETAILED REPORT OUTPUT                          *
*          PRNT   : SAME AS ABOVE(PARAMETER)                                 *
*          QHATI  : UNRELIABILITY ESTIMATES FOR EACH COMPONENT               *
*          QHTMAX : LARGEST QHATI                                            *
*          QHTUPR : UPPER LIMIT ON SYSTEM UNRELIABILITY                      *
*          QI     : INPUT UNRELIABILIY FOR EACH COMPONENT                    *
*          QINDX  : INDEX                                                    *
*          QUANTL : QUANTILE                                                 *
*          REPSHD : REPLICATIONS HEADING FORMAT NUMBER                       *
*          RHTSTR : SERIES SYSTEM RELIABILITY ESTIMATE(CLOSED FORM)          *
*          RS     : TRUE SERIES SYSTEM RELIABILITY                           *
*          RSBRDG : TRUE BRIDGE SYSTEM RELIABILITY                           *
*          RSHAT  : SERIES SYSTEM RELIABILITY ESTIMATE                       *
*          RSHTBR : BRIDGE SYSTEM RELIABILITY ESTIMATE                       *
*          SEED   : PARAMETER                                                *
*          SELCTA : SIGNIFICANCE LEVEL SELECTION                             *
*          SELCTB : SIGNIFICANCE LEVEL SELECTION                             *
*          SORT   : PARAMETER FOR ROUTINE SRND                               *
*          SUMNAI : SUM OF THE PRODUCT OF NI'S AND AI'S                      *
*          TEMP   : TEMPORARY ARRAY                                          *
*          TOTREP : TOTAL NUMBER OF PROGRAM ITERATIONS                       *
*          TRANBR : TEMPORARY ARRAY                                          *
*          TRANSQ : TEMPORARY ARRAY                                          *
*          TRANSR : TEMPORARY ARRAY                                          *
*          TRIALS : BERNOULLI TRIALS ARRAY (2-DIM)                           *
*          TRNSTR : TEMPORARY ARRAY                                          *
*          TRUQNT : TRUE QUANTILE                                            *
*          UNIRV  : UNIFORM RANDOM DEVIATES (2-DIM)                          *
*          ZFAILS : TOTAL NUMBER OF REPLICATIONS WITH ZERO FAILURES          *
*          ZFPREP : NO. OF COMPNTS. WITH ZERO FAILURES PER REPLICATION       *
*                                                                            *
******************************************************************************

      PARAMETER (KK=10,MAXALF=2,NPRNT=0)
      PARAMETER (MAXREP=1000, MAXRUN=10000, EPS=.000001)

      REAL*4 UNIRV(15,1000),TEMP(1000),QI(KK), AI(KK), AHATI(KK)
      REAL*4 QHATI(KK), NMAX, QHTMAX, ALFA(MAXALF)
      REAL*4 DF(5),AALFA(5),SUMNAI,RSHAT(MAXALF,MAXREP),RS
      REAL*4 KEY1(MAXREP),KEY2(MAXREP),KEY3(MAXREP),TRNSTR(MAXREP)
      REAL*4 DEGFR(MAXREP), QHTUPR(MAXALF,MAXREP),CHISQ(MAXALF,MAXREP)
      REAL*4 RHTSTR(MAXALF,MAXREP)
      REAL*4 DELTAR(MAXALF), TRANSQ(MAXREP),TRANSR(MAXREP),DIFF(MAXREP)
      REAL*4 DELSTR(MAXALF), NN(KK)
      REAL*4 RSHTBR(MAXALF,MAXREP),DELBRG(MAXALF),KEY4(MAXREP)
      REAL*4 TRANBR(MAXREP), RSBRDG
      REAL*4 SCALEN, MINQI, MAXQI, ZFPREP,MSTRQ

      INTEGER SEED, MULT, SORT, TRIALS(15,1000), BIGF, FI(KK), N(KK)
```

```
      INTEGER NINDX, QINDX, ERROR, REPS, SELCTA, SELCTQ, TOTREP
      INTEGER COUNT1, C1C15, REPSHD, SELCTB, ALF, FLAG, LOOP,PRNT
      INTEGER QUANTL(MAXALF), TRUQNT(MAXALF), ZFAILS, FAILS
      INTEGER INC

      DATA SEED/123457/, MULT/1/, INC/1/
      DATA AALFA/.01,.05,.9,.95,.99/, DF/1,5,10,30,40/
      DATA ALFA/.20,.050/
*     DATA MSTRQ/0.1, .05/
*     DATA AI/.20,.10,.05,.05,.05/
*     DATA AI/.20,.10,.05,.05,.05,.05,.05,.05,.05,.05/
*     DATA AI/.05,.05,.05,.05,.05,.05,.05,.05,.05,.05/
*     DATA AI/.05,.05,.05,.05,.05,.05,.05,.05,.05,.05,.05,.05,.05,
*    +.05,.05/
*     DATA AI/.1,.1,.1,.1,1,.05,.05,.2,.25,.5/
*     DATA AI/.1,.1,1.,.05,.2/
*     DATA N/100,100,100,100,100/
*     DATA N/10,6,5,2,10,4,5,6,7,2/
*     DATA N/50,50,50,50,50,50,50,50,50,50/
*     DATA N/100,100,100,100,100,100,100,100,100,100/
*     DATA N/300,300,300,300,300,300,300,300,300,300/
*     DATA N/20,20,20,20,20,20,20,20,20,20/
*     DATA N/1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,
*    +1000,1000,1000,1000/
      DATA SORT/0/

      ASSIGN 8 TO C1C15
      ASSIGN 9 TO REPSHD

*     CALL EXCMS('FILEDEF 01 DISK B OUTPUT A1 (LRECL 132 ')
*     CALL EXCMS('FILEDEF 02 DISK JNK OUTPUT A1 (LRECL 132 ')

      SELCTQ = 1
      PRNT = NPRNT

      DO 12 I=1,KK
           AI(I) = 9999.
           N(I) = 99999999
   12 CONTINUE
      READ(03,*)K,MSTRQ

      DO 11 I=1,K
      READ(03,*) AI(I),N(I)
   11 CONTINUE

***// INITIALIZE THE QHTUPR ARRAY OF UNRELIABILITY REPLICATIONS,   //***
*     RSHAT ARRAY OF ESTIMATE SYSTEM RELIABILITY REPLICATIONS          *
*     AND RHTSTR ARRAY OF EST. SYST. REL. FOR A SERIES SYST WHEN       *
***// ALL THE COMPONENT MISSION TESTS ARE EQUAL IN NUMBER          //***


      DO 172 ALF=1,MAXALF
           DO 173 REPS=1,MAXREP
                QHTUPR(ALF,REPS) = 0.
                RSHAT(ALF,REPS) = 0.
                RHTSTR(ALF,REPS) = 0.
```

```
                RSHTBR(ALF,REPS) = 0.
 173        CONTINUE
 172  CONTINUE

***// SET FLAG TO 1 IF ALL COMPONENTS HAVE SAME NO. OF MISSION TESTS****

      FLAG=1
      DO 50 I=1,K -1
           IF((N(I) - N(I+1)).NE.0) THEN
                FLAG=0
           ELSE
           END IF
  50  CONTINUE
      PRINT *, 'FLAG IS:', FLAG

***// MAIN PROGRAM OUTER LOOP START(EVERY LOOP IS ONE REPLICATION)//***

      ZFPREP = 0.
      FAILS = 0
      ZFAILS = 0
      COUNT1 = 0
      TOTREP = 0
      LOOP = 0
  10  IF(LOOP.LT.MAXREP) THEN
           IF(TOTREP.LT.MAXRUN) THEN
      SELCTA = 1
      SELCTB = 2

***// FILL ARRAY KEY(REPS) WITH INTEGERS 1 TO K TO BE USED AS OUTPUT
***// OF THE SUBROUTINE SHSORT

      DO 95 REPS=1, MAXREP
           KEY1(REPS) = REPS
           KEY2(REPS) = REPS
           KEY3(REPS) = REPS
           KEY4(REPS) = REPS
  95  CONTINUE

***// CALCULATE NMAX NOT TO PRINT LONGER THAN THE MAX SAMPLE SIZE   //***

***// CALCULATE THE MAXIMUM NO. OF TRIALS AND ITS INDEX NO.          //***

      CALL IMAX(N,K,NMAX,NINDX)


***// CALCULATE THE QI'S FROM THE GIVEN MASTER Q AND THE AI'S

      DO 115 I=1, K
           QI(I) = MSTRQ * AI(I)
 115  CONTINUE

      DO 120 I=1,15
           DO 125 J=1,500
                UNIRV(I,J) = 999.
                TRIALS(I,J) = 99999
 125       CONTINUE
```

```
 120  CONTINUE

***// CALCULATE THE SCALING FACTOR 'SCALEN'                          //***

      CALL USMNMX(QI,K,INC,MINQI,MAXQI)
      SCALEN = MAXQI / MINQI

***// DRAW UNIFORM (0,1) RV'S AND CONVERT TO BERNOULLI TRIALS        //***

      DO 130 I=1, K
           CALL SRND(SEED, TEMP, N(I), MULT, SORT)
           DO 135 J=1, N(I)
                UNIRV(I,J) = TEMP(J)
                IF (UNIRV(I,J).LE. 1 - QI(I)) THEN
                     TRIALS(I,J) = 0
                ELSE
                     TRIALS(I,J) = 1
                END IF
 135  CONTINUE
 130  CONTINUE

***// CALCULATE THE NO. OF FAILURES FOR EACH COMPONENT               //***

      DO 150 I=1, K
           FI(I) = 0
 150  CONTINUE

***// CALCULATE THE F SUB I'S AND THE GRAND TOTAL NO. OF FAILURES

      BIGF = 0
      DO 155 I=1, K
           DO 160 J=1, N(I)
                FI(I) = FI(I) + TRIALS(I,J)
 160           CONTINUE

***// CALCULATE THE QHAT SUB I'S: F SUB I'S DIVIDED BY N SUB I'S     //***

           IF(FI(I).EQ.0) THEN
                QHATI(I) = 1. / (SCALEN * N(I))
                ZFPREP = ZFPREP + 1
           ELSE
                QHATI(I) = REAL(FI(I)) / N(I)
           END IF
           BIGF = BIGF + FI(I)
 155  CONTINUE

***// FIND THE MAX OF THE INDIVIDUAL COMPONENT UNRELIABILITIES       //***

      CALL RMAX(QHATI, K, QHTMAX, QINDX)

      IF(PRNT.EQ.1) THEN
           WRITE(1,0007)
           WRITE(1,C1C15)
           WRITE(1,3334) QI
           WRITE(1,0001)
```

```
          WRITE(1,C1C15)
          DO 140 J=1,NMAX
               WRITE(1,1111) (UNIRV(I,J), I=1, K)
 140      CONTINUE
          WRITE(1,0002)
          WRITE(1,C1C15)
          DO 145 J=1,NMAX
               WRITE(1,2222) (TRIALS(I,J), I=1, K)
 145      CONTINUE
          WRITE(1,0003)
          WRITE(1,C1C15)
          WRITE(1,3333) FI
          WRITE(1,0005)
          WRITE(1,C1C15)
          WRITE(1,3335) N
          WRITE(1,0004)
          WRITE(1,C1C15)
          WRITE(1,3334) QHATI
     WRITE(1,'(/''THE MAXIMUM Q HAT SUB I IS:'', T40, F8.5)') QHTMAX
     WRITE(1,'(/''THE MAXI Q HAT SUB I IS ELMNT NO.:'',T40,I5)') QINDX
     WRITE(1,'(/''THE GRAND TOTAL NO. OF FAILURES IS:'',T40, I5)') BIGF
     ELSE
     ENDIF

***// TEST FOR A REP WITH AT LST TWO COMP WITH AT LST ONE FAILURE EACH

     DO  162 J=1, K
          NN(J) = 0
          IF(FI(J).GE.1) THEN
               COUNT1 = COUNT1 + 1
               NN(J)= N(J)
          ELSE
               NN(J) = N(J) * SCALEN
          END IF
 162 CONTINUE
     IF(COUNT1.EQ.0) THEN
          ZFAILS = ZFAILS + 1
     ELSE
          FAILS = FAILS + 1
     END IF
     IF(COUNT1.GE.2) THEN
          LOOP = LOOP + 1
          TOTREP = TOTREP + 1
     ELSE
          TOTREP = TOTREP + 1
          GO TO 10
     END IF

***// CALCULATE THE AHAT SUB I'S (WEIGHT ESTIMATES)                    //***

     SUMNAI = 0.
     DO 165 I=1, K
          AHATI(I) = QHATI(I) / QHTMAX
*    +(FI(L),L=1,K)
*    +(NN(L),L=1,K)
          SUMNAI = SUMNAI + NN(I) * AHATI(I)
```

```
 165  CONTINUE
      IF(PRNT.EQ.1) THEN
           WRITE(1,0006)
           WRITE(1,C1C15)
           WRITE(1,3334) AHATI
      ELSE
      END IF

***// CALCULATE 1 REPLICATION OF UPPR ALFA C.L. ON SYSTEM RELIABILITY

      DEGFR(LOOP) = 2 * (1 + BIGF)

      DO 170 ALF=1, MAXALF
           CALL MDCHI(1 - ALFA(ALF),DEGFR(LOOP),CHISQ(ALF,LOOP), ERROR)
           QHTUPR(ALF,LOOP) = CHISQ(ALF,LOOP) / (2 * SUMNAI)
           IF(FLAG.EQ.1) THEN
                RHTSTR(ALF,LOOP) = 1 -(CHISQ(ALF,LOOP) / REAL(2*N(1)))
           ELSE
           END IF
*    +     (ALF,LOOP), ALFA(ALF)

***// CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR COMPNTS. IN SERIES

           CALL RHTSRS(QHTUPR(ALF,LOOP), AHATI,K, RSHAT(ALF,LOOP))
*    +T40,F8.5)') RSHAT(ALF,LOOP)

***// CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR BRIDGE STRUCTURE ***

           CALL RHTBRG(QHTUPR(ALF,LOOP),AHATI,K,RSHTBR(ALF,LOOP))

 170  CONTINUE

***// THIS ELSE AND ENDIF ARE FOR THE TEST AGAINST MAXRUN          //***
      ELSE
           WRITE(1,'('' '',/''PROGRAM EXCEEDED THE MAX NO. OF RUNS'',
     +'' ALLOWED OF:'',I6)') TOTREP
      GOTO 9999

      END IF
      GOTO 10
      END IF

***// SORT THE ARRAYS OF SYSTEM UNRELIABILITIES(1 FOR EACH CONF. LEVEL)

      DO 700 ALF=1, MAXALF
           DO 800 REPS=1, MAXREP
                TRANSQ(REPS) = QHTUPR(ALF,REPS)
                TRANSR(REPS) = RSHAT(ALF,REPS)
                TRNSTR(REPS) = RHTSTR(ALF,REPS)
                TRANBR(REPS) = RSHTBR(ALF,REPS)
 800       CONTINUE
           CALL SHSORT(TRANSQ,KEY1,MAXREP)
           CALL SHSORT(TRANSR,KEY2,MAXREP)
           CALL SHSORT(TRNSTR,KEY3,MAXREP)
           CALL SHSORT(TRANBR,KEY4,MAXREP)
           DO 900 REPS=1, MAXREP
```

```
            QHTUPR(ALF,REPS) = TRANSQ(REPS)
            RSHAT(ALF,REPS) = TRANSR(REPS)
            RHTSTR(ALF,REPS) = TRNSTR(REPS)
            RSHTBR(ALF,REPS) = TRANBR(REPS)
900        CONTINUE
700   CONTINUE

***// PRINT OUTPUT REPORT HEADINGS                                          //***

      WRITE(1,6666)
      WRITE(1,6667) MAXREP
      WRITE(1,6668) K
      WRITE(1,6770) SCALEN
      WRITE(1,6669)
      IF(K.EQ.5) THEN
          WRITE(1,6699)
      ELSE
      END IF
      WRITE(1,6670) MSTRQ
      WRITE(1,6671)
      WRITE(1,C1C15)
      WRITE(1,3334) AI
      WRITE(1,0007)
      WRITE(1,C1C15)
      WRITE(1,3334) QI
      WRITE(1,0005)
      WRITE(1,C1C15)
      WRITE(1,3335) N
      WRITE(1,6674)
      WRITE(2,6666)
      WRITE(2,6667) MAXREP
      WRITE(2,6668) K
      WRITE(2,6770) SCALEN
      WRITE(2,6669)
      IF(K.EQ.5) THEN
          WRITE(1,6699)
      ELSE
      END IF
      WRITE(2,6670) MSTRQ
      WRITE(2,6671)
      WRITE(2,C1C15)
      WRITE(2,3334) AI
      WRITE(2,0007)
      WRITE(2,C1C15)
      WRITE(2,3334) QI
      WRITE(2,0005)
      WRITE(2,C1C15)
      WRITE(2,3335) N
      WRITE(2,6674)
      WRITE(2,'(''SORTED RSHAT 1  IS:'',/15(F8.5))')
     +(RSHAT(1,REPS), REPS=1, MAXREP)
      WRITE(2,'(''SORTED RSHAT 2  IS:'',/15(F8.5))')
     +(RSHAT(2,REPS), REPS=1, MAXREP)
      IF(FLAG.EQ.1) THEN
          WRITE(2,'(''SORTED RHTSTR 1  IS:'',/10(F8.5))')
```

```
     +(RHTSTR(1,REPS), REPS=1, MAXREP)
           WRITE(2,'(''SORTED RHTSTR 2  IS:'',/10(F8.5))')
     +(RHTSTR(2,REPS), REPS=1, MAXREP)
      ELSE
      END IF
      IF(K.EQ.5) THEN
           WRITE(2,'(''SORTED RSHTBR 1  IS:'',/10(F8.5))')
     +(RSHTBR(1,REPS), REPS=1, MAXREP)
           WRITE(2,'(''SORTED RSHTBR 2  IS:'',/10(F8.5))')
     +(RSHTBR(2,REPS), REPS=1, MAXREP)
      ELSE
      END IF

***// COMPUTE THE VALUE RS OF THE TRUE SYSTEM REL. FNCTN.(SERIES SYSTEM)
***// AND FOR THE 5-COMPONENT BRIDGE STRUCTURE

      CALL RSRS(QI,K,RS)
      WRITE(1,'('' '',////''THE TRUE SERIES SYSTEM '',
     +''RELIABILITY VALUE IS:'',T51,F8.5)') RS
      CALL RBRIDG(QI,K,RSBRDG)
      IF(K.EQ.5) THEN
      WRITE(1,'('' '',////''THE TRUE BRIDGE STRUCTURE '',
     +''RELIABILITY VALUE IS:'',T51,F8.5)') RSBRDG
      ELSE
      END IF
      WRITE(1,6675)

***// COMPUTE THE DIFFERENCE 'DELTAR' BTWN. RS AND RSHAT OF THE THEO
***// RETICAL QUANTILE GIVEN BY ALFA(MUST USE SORTED RSHAT ARRAY)

      IF(FLAG.EQ.1) THEN
           WRITE(1,5755)
      ELSE
      END IF
      DO 450 ALF=1, MAXALF
           QUANTL(ALF) = MAXREP * (1 - ALFA(ALF))
           DELTAR(ALF) = RS - RSHAT(ALF,QUANTL(ALF))
           DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))
           IF(FLAG.EQ.1) THEN
                DELSTR(ALF) = RS - RHTSTR(ALF,QUANTL(ALF))
                WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
                WRITE(1,5656) RHTSTR(ALF,QUANTL(ALF))
                WRITE(1,5657) DELSTR(ALF)
           ELSE
           END IF
           IF(K.EQ.5) THEN
                DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))
                WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
                WRITE(1,5666) RSHTBR(ALF,QUANTL(ALF))
                WRITE(1,5667) DELBRG(ALF)
           ELSE
           END IF
           WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
           WRITE(1,5556) RSHAT(ALF,QUANTL(ALF))
           WRITE(1,5557) DELTAR(ALF)
  450 CONTINUE
```

```
      PRINT *, 'QUANTL(1) IS:', QUANTL(1)
      PRINT *, 'QUANTL(2) IS:', QUANTL(2)

***// FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE //***
***//               ********* RSHAT ********                       //***

      WRITE(1,6676)
      DO 400 ALF=1,MAXALF
      TRUQNT(ALF) = 0
           DO 500 REPS=1, MAXREP
                DIFF(REPS) = RS - RSHAT(ALF,REPS)
 500       CONTINUE
           DO 600 REPS=1, MAXREP
                IF(ABS(DIFF(REPS)).LE.EPS) THEN
                     TRUQNT(ALF) = REPS
                     WRITE(1,'('' '',/''TRUE CONFIDENCE LIMIT IS:'',
     +               F8.4)')
     +               (TRUQNT(ALF) / REAL(MAXREP)) * 100.
                     GO TO 620
                ELSEIF(DIFF(REPS).LT.0.) THEN
                     TRUQNT(ALF) = REPS
                     GO TO 610
                ELSE
                END IF
 600       CONTINUE
 610       IF(TRUQNT(ALF).EQ.0.) THEN
                WRITE(1,4443) ALFA(ALF)
                WRITE(1,'('' '',/''THE SMALLEST'',
     +          '' DIFFERENCE BETWEEN RS AND RSHAT IS:'',F10.5)') DIFF(
     +          MAXREP)
           ELSEIF(TRUQNT(ALF).EQ.1.) THEN
                WRITE(1,4442) ALFA(ALF)
                WRITE(1,'('' '',/''ALL RSHAT'',
     +           '' ARE GREATER THAN RS'')')
           ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))
     +     THEN
                WRITE(1,4444) ALFA(ALF),
     +          (TRUQNT(ALF) / REAL(MAXREP)) * 100.
                WRITE(1,4445) RSHAT(ALF,TRUQNT(ALF))
                WRITE(1,4446)
           ELSE
                WRITE(1,4444) ALFA(ALF),
     +          ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
                WRITE(1,4445) RSHAT(ALF,TRUQNT(ALF)-1)
                WRITE(1,4447)
 620       END IF
 400  CONTINUE


***// FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE //***
***//          ********** RSHTBR (BRIDGE) ********                //***

      IF(K.EQ.5) THEN
      DO 401 ALF=1,MAXALF
      TRUQNT(ALF) = 0
           DO 501 REPS=1, MAXREP
```

```
             DIFF(REPS) = RSBRDG - RSHTBR(ALF,REPS)
 501      CONTINUE
          DO 601 REPS=1, MAXREP
             IF(ABS(DIFF(REPS)).LE.EPS) THEN
                TRUQNT(ALF) = REPS
                WRITE(1,'('' '',/''TRUE CONFIDENCE LIMIT IS:'',
     +          F8.4)')
     +          (TRUQNT(ALF) / REAL(MAXREP)) * 100.
                GO TO 621
             ELSEIF(DIFF(REPS).LT.0.) THEN
                TRUQNT(ALF) = REPS
                GO TO 611
             ELSE
             END IF
 601      CONTINUE
 611      IF(TRUQNT(ALF).EQ.0.) THEN
             WRITE(1,4443) ALFA(ALF)
             WRITE(1,'('' '',/''THE SMALLEST'',
     +       '' DIFFERENCE BETWEEN RSBRDG AND RSHTBR IS:'',
     +       F10.5)') DIFF(MAXREP)
          ELSEIF(TRUQNT(ALF).EQ.1.) THEN
             WRITE(1,4442) ALFA(ALF)
             WRITE(1,'('' '',/''ALL RSHTBR'',
     +        '' ARE GREATER THAN RSBRDG'')')
          ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))
     +    THEN
             WRITE(1,4444) ALFA(ALF),
     +       (TRUQNT(ALF) / REAL(MAXREP)) * 100.
             WRITE(1,4449) RSHTBR(ALF,TRUQNT(ALF))
             WRITE(1,4446)
          ELSE
             WRITE(1,4444) ALFA(ALF),
     +       ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
             WRITE(1,4449) RSHTBR(ALF,TRUQNT(ALF)-1)
             WRITE(1,4447)
 621      END IF
 401  CONTINUE
      ELSE
      END IF


***// FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL.  ESTIMATE //***
***//                   ********* RHTSTR *********                //***

      IF(FLAG.EQ.1) THEN
      DO 4400 ALF=1,MAXALF
      TRUQNT(ALF) = 0
          DO 5500 REPS=1, MAXREP
             DIFF(REPS) = RS - RHTSTR(ALF,REPS)
 5500     CONTINUE
          DO 6600 REPS=1, MAXREP
             IF(ABS(DIFF(REPS)).LE.EPS) THEN
                TRUQNT(ALF) = REPS
                WRITE(1,'('' '',/''TRUE CONFIDENCE LIMIT IS:'',
     +          F8.4)')
     +          (TRUQNT(ALF) / REAL(MAXREP)) * 100.
```

```
                    GO TO 6620
               ELSEIF(DIFF(REPS).LT.0.) THEN
                    TRUQNT(ALF) = REPS
                    GO TO 6610
               ELSE
               END IF
6600      CONTINUE
6610      IF(TRUQNT(ALF).EQ.0.) THEN
               WRITE(1,4443) ALFA(ALF)
               WRITE(1,'('' '',/''THE SMALLEST'',
     +         '' DIFFERENCE BETWEEN RS AND RHTSTR IS:'',
     +         F9.5)') DIFF(MAXREP)
          ELSEIF(TRUQNT(ALF).EQ.1.) THEN
               WRITE(1,4442) ALFA(ALF)
               WRITE(1,'('' '',/''ALL RHTSTR'',
     +         '' ARE GREATER THAN RS'')')
          ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))
     +    THEN
               WRITE(1,4444) ALFA(ALF),
     +         (TRUQNT(ALF) / REAL(MAXREP)) * 100.
               WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF))
               WRITE(1,4446)
          ELSE
               WRITE(1,4444) ALFA(ALF),
     +         ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
               WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF)-1)
               WRITE(1,4447)
6620      END IF
4400 CONTINUE
     ELSE
     END IF

***// PRINT THE ARRAYS PERTINENT TO THE OUPUT OF EACH REPLICATION  //***

     IF(PRNT.EQ.1) THEN
     I = 1
185  WRITE(1,REPSHD) ALFA(SELCTA), ALFA(SELCTA),
    +ALFA(SELCTB),ALFA(SELCTB),ALFA(SELCTA),ALFA(SELCTA),ALFA(SELCTB),
    +ALFA(SELCTB)
175  IF(I.GE.(MAXREP + 1)) THEN
          GOTO 180
     ELSE
          IF( (I.EQ.71).OR.(I.EQ.211).OR.(I.EQ.351).OR.(I.EQ.491).OR.
     +    (I.EQ.631).OR.(I.EQ.771).OR.(I.EQ.911).OR.(I.EQ.1051) ) THEN
               I = I + 70
               WRITE(1,'(''+'')')
               GOTO 185
          ELSE

          WRITE(1,3336) I, INT(DEGFR(I)), CHISQ(1,I), QHTUPR(1,I),
     +    CHISQ(2,I), QHTUPR(2,I)
          END IF
          IF(I + 70.LE.MAXREP) THEN
               WRITE(1,3337) I+70,INT(DEGFR(I+70)),CHISQ(1,I+70),
     +         QHTUPR(1,I+70),CHISQ(2,I+70),QHTUPR(2,I+70)
          ELSE
```

```
          END IF
      I = I + 1
      GOTO 175
180   END IF
      ELSE
      ENDIF
9999  WRITE(1,'(''THE TOTAL NO OF REPS WAS:'',I8)') TOTREP
      WRITE(1,'(''THE TOTAL NO OF EFFECTIVE REPS WAS:'',I8)') LOOP
      WRITE(1,'(''THE TOTAL NO OF NO FAILURE RUNS WAS:'',I8)') ZFAILS
      WRITE(1,'(''THE TOTAL NO OF RUNS WITH FAILURES WAS:'',I8)') FAILS
      WRITE(1,'(''AVG NO OF COMPONENTS WITH NO FAILURES PER REP WAS:'',
     +F5.2)') ZFPREP / MAXREP
0008  FORMAT (/ 3X,'C 1',5X,'C 2',
     +5X,'C 3',5X,'C 4',5X,'C 5',5X,'C 6',5X,'C 7',5X,
     +'C 8',5X,'C 9',5X,'C 10',4X,'C 11',4X,
     +'C 12',4X,'C 13',4X,'C 14',4X,'C 15')
0009  FORMAT(/1X,'REP NO',2X,'DF',1X,'CHISQR(',F4.3,')',1X,
     +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'QHTUPR(',F4.3,')',
     +2X,'REP NO',2X,'DF',1X,'CHISQR(',F4.3,')',1X,
     +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'QHTUPR(',F4.3,')'/)
0001  FORMAT (///'UNIFORM RANDOM DEVIATES ARE:')
0002  FORMAT (///'BERNOULLI TRIALS ARE:')
0003  FORMAT (///'TOTAL NO. OF FAILURES FOR EACH COMPONENT:')
0004  FORMAT (///'ESTIMATED UNRELIABILITY FOR EACH COMPONENT:')
0005  FORMAT (///'TOTAL NUMBER OF MISSION TESTS:')
0006  FORMAT (///'ESTIMATED WEIGHTS FOR EACH COMPONENT:')
0007  FORMAT (///'Q I FOR EACH COMPONENT:')
1111  FORMAT (15F8.5)

2222  FORMAT (/1X,15(I4,4X))
3333  FORMAT (/1X,15(I4,4X))
3334  FORMAT (/15F8.5)
3335  FORMAT (/1X,15(I4,4X))
3336  FORMAT (T3,I4,T9,I3,T13,F11.5,T27,F8.5,T39,F11.5,T53,F8.5)
3337  FORMAT ('+',T67,I4,T73,I3,T77,F11.5,T91,F8.5,T103,F11.5,T117,F8.5)
4442  FORMAT (' ',///'THE RESULTING (1 - ',F4.3,') CONFIDENCE ',
     +'LIMIT IS:',T50,' 00.000 ')
4443  FORMAT (' ',///'THE RESULTING (1 - ',F4.3,') CONFIDENCE ',
     +'LIMIT IS:',T50,'100.0000')
4444  FORMAT (' ',///'THE RESULTING (1 - ',F4.3,') CONFIDENCE ',
     +'LIMIT IS:',T50,F8.4)
4445  FORMAT (' ',/'THE RSHAT VALUE CLOSEST TO RS IS: ',T51,F8.5)
4446  FORMAT (' ',/'(FIRST NEGATIVE DIFFERENCE)')
4447  FORMAT (' ',/'(ELEMENT PRECEEDING FIRST NEGATIVE DIFFERENCE)')
4448  FORMAT (' ',/'THE RHTSTR VALUE CLOSEST TO RS IS: ',T51,F8.5)
4449  FORMAT (' ',/'THE RSHTBR VALUE CLOSEST TO RSBRDG IS: ',T51,F8.5)
5555  FORMAT (' ',///'THE ',I4,'(1-',F4.3,') QUANTILE IS:',T49,F8.3)
5556  FORMAT (' ',/'THE VALUE OF RSHAT FOR THAT QUANTILE IS:',T51,F8.5)
5557  FORMAT (' ',/'THE DIFFERENCE(RS - RSHAT) IS:',T51,F8.5)
5656  FORMAT (' ',/'THE VALUE OF RHTSTR FOR THAT QUANTILE IS:',T51,F8.5)
5666  FORMAT (' ',/'THE VALUE OF RSHTBR FOR THAT QUANTILE IS:',T51,F8.5)
5657  FORMAT (' ',/'THE DIFFERENCE(RS - RHTSTR) IS:',T51,F8.5)
5667  FORMAT (' ',/'THE DIFFERENCE(RS - RSHTBR) IS:',T51,F8.5)
5755  FORMAT (' ',///'SINCE THE NO. OF MISSION TESTS IS THE SAME FOR',
     +' ALL COMPONENTS THE CLOSED FORM SERIES SYSTEM RELIABILITY ',
     +'''RHTSTR'' IS COMPUTED')
```

```
6666 FORMAT ('+','***********************************************',
    +'* RUN INPUT SETTINGS FOR ADJUSTED SUM OF THE NI AI ***********',
    +'*********************')
6667 FORMAT (' ',//'NUMBER OF REPLICATIONS:',T50,I4)
6668 FORMAT (' ',//'NUMBER OF COMPONENTS:',T50,I4)
6669 FORMAT (' ',//'SYSTEM RELIABILITY FUNCTION:',T50,'SERIES')
6699 FORMAT (' ',//'SYSTEM RELIABILITY FUNCTION:',T50,'BRIDGE')
6670 FORMAT (' ',//'MASTER UNRELIABILITY USED:',T50,F8.5)
6770 FORMAT (' ',//'ZERO FAILURE SCALING FACTOR:',T50,F5.2)
6671 FORMAT (' ',//'INPUT WEIGHTS(A SUB I''S):')
6674 FORMAT ('+',///'***********************************************',
    +'**R U N  R E S U L T S****************************************',
    +'**************')
6675 FORMAT ('+',///'************************************************',
    +' ESTIMATE ERRORS *********************************************',
    +'**************')
6676 FORMAT ('+',///'***********************************************',
    +' TRUE CONFIDENCE LIMITS ***************************************',
    +'**************')
     END
```

APPENDIX C. INTERVAL ESTIMATING PROCEDURE 3

```
      PROGRAM ZFYSCY
*************************************************************************
*                                                                       *
*      TITLE: BINOMIAL INTERVAL ESTIMATION PROCEDURE                   *
*             ZERO FAILURES DISALLOWED; WITH SCALING                   *
*     AUTHOR: E. F. BELLINI, LT, USN                                   *
*       DATE: NOV 89                                                   *
*                                                                       *
*     THIS PROGRAM COMPUTES THE TRUE CONFIDENCE LEVEL FOR THE ESTIMATE *
*     RELIABILITY OF A SERIES AND BRIDGE SYSTEM GIVEN THE RELIABILITY  *
*     OF THEIR COMPONENTS                                              *
*                                                                       *
*     IN ITS PRESENT CONFIGURATION THIS PROGRAM IS SET UP TO RUN 12    *
*     TIMES EACH TIME PRODUCING 1000 REPLICATIONS USING A DIFFERENT    *
*     SET OF INPUT DATA.  RUN THE PROGRAM FROM CMS BY TYPING 'B1 EXEC'.*
*     THE REXX EXEC PROGRAM                                            *
*     'B1' CALLS THE INPUT FILES TO BE READ AND NAMES THE 12 OUTPUT    *
*     FILES RESULTING FROM THE 12 CONSECUTIVE RUNS.  BY EDITING THE    *
*     INDEX COUNTERS I, J, K OF THE 'B1' EXEC ONE CAN RUN ANY USER-    *
*     SPECIFIC RUN FROM JUST ONE RUN TO ALL 12.                        *
*                                                                       *
*     VARIABLES USED                                                   *
*                                                                       *
*           AHATI  : WEIGHT ESTIMATES FOR EACH COMPONENT               *
*           AI     : INPUT WEIGHTS FOR EACH COMPONENT                  *
*           ALFA   : LEVELS OF SIGNIFICANCE                            *
*           BIGF   : TOTAL NO. OF FAILURES FOR EACH REPLICATION        *
*           CHISQ  : CHI-SQUARE RANDOM VARIABLE VALUE                  *
*           COUNT1 : COUNTS THE NO. OF COMPONENTS WITH FAILURES        *
*           C1C15  : FORMAT LABEL                                      *
*           DEGFR  : DEGREES OF FREEDOM                                *
*           DELBRG : DIFFERENCE FOR BRIDGE SYSTEM                      *
*           DELSTR : DIFFERENCE FOR SERIES SYSTEM- CLOSED FORM         *
*           DELTAR : DIFFERENCE FOR SERIES SYSTEM                      *
*           DIFF   : DIFFERENCE (TRUE REL. - ESTIMATED REL.)           *
*           EPS    : SMALL QUANTITY(CONSTANT)                          *
*           ERROR  : PARAMETER FOR IMSL ROUTINE                        *
*           FAILS  : COUNTS NO. OF REPLICATIONS WITH AT LST. 1 FAILURE *
*           FI     : NO. OF FAILURES FOR EACH COMPONENT(ALL MISSION TST)*
*           FLAG   : 1 IF ALL COMP. HAVE SAME NO. OF MISSION TESTS     *
*           INC    : INCREMENT STEP SIZE FOR ROUTINE USMNMX            *
*           KEY1   : ARRAY OF INDECES FOR ROUTINE SHSORT               *
*           KEY2   : ARRAY OF INDECES FOR ROUTINE SHSORT               *
*           KEY3   : ARRAY OF INDECES FOR ROUTINE SHSORT               *
*           KEY4   : ARRAY OF INDECES FOR ROUTINE SHSORT               *
*           KK     : ARRAY SIZING PARAMETER FOR THE MAX NO OF COMPONENTS*
*           LOOP   : COUNTS NO. OF REPLICATION PERFORMED               *
*           MAXALF : MAX NO. OF SIGNIFICANT LEVELS DESIRED(ARRAY SIZING)*
*           MAXREP : MAX NO. OF REPLICATIONS                           *
*           MAXRUN : MAX NO. OF PROGRAM ITERATIONS ALLOWED             *
*           MSTRQ  : MASTER UNRELIABILITY(USED WITH AI'S TO CALC. QI'S) *
```

```
*          MULT   : MULTIPLIER FOR RANDOM NO. GENERATOR SRND                *
*          N      : NO. OF MISSION TEST FOR EACH COMPONENT                  *
*          NIMAX  : MAX NO. OF MISSION TESTS                                *
*          NIMIM  : MIN NO. OF MISSION TESTS                                *
*          NINDX  : INDEX NO. OF MAX NO. OF MISSION TESTS                   *
*          NIREAL : NO. OF MISSION TESTS TRANSFORMED TO REAL                *
*          NMAX   : MAX NO. OF MISSION TESTS FOR OUTPUT CONTROL             *
*          NPRNT  : FLAG FOR DETAILED REPORT OUTPUT                         *
*          PRNT   : SAME AS ABOVE(PARAMETER)                                *
*          QHATI  : UNRELIABILITY ESTIMATES FOR EACH COMPONENT              *
*          QHTMAX : LARGEST QHATI                                           *
*          QHTUPR : UPPER LIMIT ON SYSTEM UNRELIABILITY                     *
*          QI     : INPUT UNRELIABILIY FOR EACH COMPONENT                   *
*          QINDX  : INDEX                                                   *
*          QUANTL : QUANTILE                                                *
*          REPSHD : REPLICATIONS HEADING FORMAT NUMBER                      *
*          RHTSTR : SERIES SYSTEM RELIABILITY ESTIMATE(CLOSED FORM)         *
*          RS     : TRUE SERIES SYSTEM RELIABILITY                          *
*          RSBRDG : TRUE BRIDGE SYSTEM RELIABILITY                          *
*          RSHAT  : SERIES SYSTEM RELIABILITY ESTIMATE                      *
*          RSHTBR : BRIDGE SYSTEM RELIABILITY ESTIMATE                      *
*          SEED   : PARAMETER                                               *
*          SELCTA : SIGNIFICANCE LEVEL SELECTION                            *
*          SELCTB : SIGNIFICANCE LEVEL SELECTION                            *
*          SORT   : PARAMETER FOR ROUTINE SRND                              *
*          SUMNAI : SUM OF THE PRODUCT OF NI'S AND AI'S                     *
*          TEMP   : TEMPORARY ARRAY                                         *
*          TOTREP : TOTAL NUMBER OF PROGRAM ITERATIONS                      *
*          TRANBR : TEMPORARY ARRAY                                         *
*          TRANSQ : TEMPORARY ARRAY                                         *
*          TRANSR : TEMPORARY ARRAY                                         *
*          TRIALS : BERNOULLI TRIALS ARRAY (2-DIM)                          *
*          TRNSTR : TEMPORARY ARRAY                                         *
*          TRUQNT : TRUE QUANTILE                                           *
*          UNIRV  : UNIFORM RANDOM DEVIATES (2-DIM)                         *
*          ZFAILS : TOTAL NUMBER OF REPLICATIONS WITH ZERO FAILURES         *
*          ZFPREP : NO. OF COMPNTS. WITH ZERO FAILURES PER REPLICATION      *
*                                                                           *
*****************************************************************************

      PARAMETER (KK=10,MAXALF=2,NPRNT=0)
      PARAMETER (MAXREP=1000, MAXRUN=10000, EPS=.000001)

      REAL*4 UNIRV(15,1000),TEMP(1000),QI(KK), AI(KK), AHATI(KK)
      REAL*4 QHATI(KK), NMAX, NNMAX, QHTMAX, ALFA(MAXALF)
      REAL*4 DF(5),AALFA(5),SUMNAI,RSHAT(MAXALF,MAXREP),RS
      REAL*4 KEY1(MAXREP),KEY2(MAXREP),KEY3(MAXREP),TRNSTR(MAXREP)
      REAL*4 DEGFR(MAXREP), QHTUPR(MAXALF,MAXREP),CHISQ(MAXALF,MAXREP)
      REAL*4 RHTSTR(MAXALF,MAXREP)
      REAL*4 DELTAR(MAXALF), TRANSQ(MAXREP),TRANSR(MAXREP),DIFF(MAXREP)
      REAL*4 DELSTR(MAXALF), NN(KK)
      REAL*4 RSHTBR(MAXALF,MAXREP),DELBRG(MAXALF),KEY4(MAXREP)
      REAL*4 TRANBR(MAXREP), RSBRDG
      REAL*4 SCALEN, MINQI, MAXQI, ZFPREP,MSTRQ

      INTEGER SEED, MULT, SORT, TRIALS(15,1000), BIGF, FI(KK), N(KK)
```

```
      INTEGER NINDX, QINDX, ERROR, REPS, SELCTA, SELCTQ, TOTREP
      INTEGER COUNT1, C1C15, REPSHD, SELCTB, ALF, FLAG, LOOP,PRNT
      INTEGER QUANTL(MAXALF), TRUQNT(MAXALF), ZFAILS, FAILS
      INTEGER INC

      DATA SEED/123457/, MULT/1/, INC/1/
      DATA AALFA/.01,.05,.9,.95,.99/, DF/1,5,10,30,40/
      DATA ALFA/.20,.050/
*     DATA MSTRQ/0.1, .05/
*     DATA AI/.20,.10,.05,.05,.05/
*     DATA AI/.20,.10,.05,.05,.05,.05,.05,.05,.05,.05/
*     DATA AI/.05,.05,.05,.05,.05,.05,.05,.05,.05,.05/
*     DATA AI/.05,.05,.05,.05,.05,.05,.05,.05,.05,.05,.05,.05,.05,
*    +.05,.05/
*     DATA AI/.1,.1,.1,.1,1,.05,.05,.2,.25,.5/
*     DATA AI/.1,.1,1.,.05,.2/
*     DATA N/100,100,100,100,100/
*     DATA N/10,6,5,2,10,4,5,6,7,2/
*     DATA N/50,50,50,50,50,50,50,50,50,50/
*     DATA N/100,100,100,100,100,100,100,100,100,100/
*     DATA N/300,300,300,300,300,300,300,300,300,300/
*     DATA N/20,20,20,20,20,20,20,20,20,20/
*     DATA N/1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,
*    +1000,1000,1000,1000/
      DATA SORT/0/

      ASSIGN 8 TO C1C15
      ASSIGN 9 TO REPSHD

*     CALL EXCMS('FILEDEF 01 DISK B OUTPUT A1 (LRECL 132 ')
*     CALL EXCMS('FILEDEF 02 DISK JNK OUTPUT A1 (LRECL 132 ')

      SELCTQ = 1
      PRNT = NPRNT

      DO 12 I=1,KK
           AI(I) = 9999.
           N(I) = 99999999
   12 CONTINUE
      READ(03,*)K,MSTRQ

      DO 11 I=1,K
      READ(03,*) AI(I),N(I)
   11 CONTINUE

***// INITIALIZE THE QHTUPR ARRAY OF UNRELIABILITY REPLICATIONS,      //***
*     RSHAT ARRAY OF ESTIMATE SYSTEM RELIABILITY REPLICATIONS              *
*     AND RHTSTR ARRAY OF EST. SYST. REL. FOR A SERIES SYST WHEN           *
***// ALL THE COMPONENT MISSION TESTS ARE EQUAL IN NUMBER             //***


      DO 172 ALF=1,MAXALF
           DO 173 REPS=1,MAXREP
                QHTUPR(ALF,REPS) = 0.
                RSHAT(ALF,REPS) = 0.
                RHTSTR(ALF,REPS) = 0.
```

```
                 RSHTBR(ALF,REPS) = 0.
 173        CONTINUE
 172   CONTINUE

***// SET FLAG TO 1 IF ALL COMPONENTS HAVE SAME NO. OF MISSION TESTS****

       FLAG=1
       DO 50 I=1,K -1
            IF((N(I) - N(I+1)).NE.0) THEN
                 FLAG=0
            ELSE
            END IF
  50   CONTINUE
       PRINT *, 'FLAG IS:', FLAG
***// MAIN PROGRAM OUTER LOOP START(EVERY LOOP IS ONE REPLICATION)//***

       ZFPREP = 0.
       FAILS = 0
       ZFAILS = 0
       COUNT1 = 0
       TOTREP = 0
       LOOP = 0
  10   IF(LOOP.LT.MAXREP) THEN
            IF(TOTREP.LT.MAXRUN) THEN
       SELCTA = 1
       SELCTB = 2

***// FILL ARRAY KEY(REPS) WITH INTEGERS 1 TO K TO BE USED AS OUTPUT
***// OF THE SUBROUTINE SHSORT

       DO 95 REPS=1, MAXREP
            KEY1(REPS) = REPS
            KEY2(REPS) = REPS
            KEY3(REPS) = REPS
            KEY4(REPS) = REPS
  95   CONTINUE

***// CALCULATE NMAX NOT TO PRINT LONGER THAN THE MAX SAMPLE SIZE

***// CALCULATE THE MAXIMUM NO. OF TRIALS AND ITS INDEX NO.          //***

       CALL IMAX(N,K,NMAX,NINDX)


***// CALCULATE THE QI'S FROM THE GIVEN MASTER Q AND THE AI'S        //***

       DO 115 I=1, K
            QI(I) = MSTRQ * AI(I)
 115   CONTINUE

       DO 120 I=1,15
            DO 125 J=1,500
                 UNIRV(I,J) = 999.
                 TRIALS(I,J) = 99999
 125        CONTINUE
 120   CONTINUE
```

```
***// CALCULATE THE SCALING FACTOR 'SCALEN'                                  //***

      CALL USMNMX(QI,K,INC,MINQI,MAXQI)
      SCALEN = MAXQI / MINQI
***// DRAW UNIFORM (0,1) RV'S AND CONVERT TO BERNOULLI TRIALS //***

      DO 130 I=1, K
           CALL SRND(SEED, TEMP, N(I), MULT, SORT)
           DO 135 J=1, N(I)
                UNIRV(I,J) = TEMP(J)
                IF (UNIRV(I,J).LE. 1 - QI(I)) THEN
                     TRIALS(I,J) = 0
                ELSE
                     TRIALS(I,J) = 1
                END IF
 135  CONTINUE
 130  CONTINUE


***// CALCULATE THE NO. OF FAILURES FOR EACH COMPONENT                      //***

      DO 150 I=1, K
           FI(I) = 0
 150  CONTINUE

***// CALCULATE THE F SUB I'S AND THE GRAND TOTAL NO. OF FAILURES

      BIGF = 0
      DO 155 I=1, K
           DO 160 J=1, N(I)
                FI(I) = FI(I) + TRIALS(I,J)
 160         CONTINUE

***// CALCULATE THE QHAT SUB I'S: F SUB I'S DIVIDED BY N SUB I'S
           IF(FI(I).EQ.0) THEN
                QHATI(I) = 1. / (SCALEN * N(I))
                ZFPREP = ZFPREP + 1
           ELSE
                QHATI(I) = REAL(FI(I)) / N(I)
           END IF
           BIGF = BIGF + FI(I)
 155  CONTINUE

***// FIND THE MAX OF THE INDIVIDUAL COMPONENT UNRELIABILITIES
      CALL RMAX(OHATI, K, QHTMAX, QINDX)

      IF(PRNT.EQ.1) THEN
           WRITE(1,0007)
           WRITE(1,C1C15)
           WRITE(1,3334) QI
           WRITE(1,0001)
           WRITE(1,C1C15)
           DO 140 J=1,NMAX
                WRITE(1,1111) (UNIRV(I,J), I=1, K)
 140       CONTINUE
```

```
          WRITE(1,0002)
          WRITE(1,C1C15)
          DO 145 J=1,NMAX
               WRITE(1,2222) (TRIALS(I,J), I=1, K)
 145      CONTINUE
          WRITE(1,0003)
          WRITE(1,C1C15)
          WRITE(1,3333) FI
          WRITE(1,0005)
          WRITE(1,C1C15)
          WRITE(1,3335) N
          WRITE(1,0004)
          WRITE(1,C1C15)
          WRITE(1,3334) QHATI
     WRITE(1,'(/''THE MAXIMUM Q HAT SUB I IS:'', T40, F8.5)') QHTMAX
     WRITE(1,'(/''THE MAXI Q HAT SUB I IS ELMNT NO.:'',T40,I5)') QINDX
     WRITE(1,'(/''THE GRAND TOTAL NO. OF FAILURES IS:'',T40, I5)') BIGF
     ELSE
     ENDIF

***// TEST FOR A REP WITH AT LST TWO COMP WITH AT LST ONE FAILURE EACH
     DO  162 J=1, K
          NN(J) = 0
          IF(FI(J).GE.1) THEN
               COUNT1 = COUNT1 + 1
               NN(J)= N(J)
          ELSE
               NN(J) = N(J) * SCALEN
          END IF
 162 CONTINUE
     IF(COUNT1.EQ.0) THEN
          ZFAILS = ZFAILS + 1
     ELSE
          FAILS = FAILS + 1
     END IF
     IF(COUNT1.GE.0) THEN
          LOOP = LOOP + 1
          TOTREP = TOTREP + 1
     ELSE
          TOTREP = TOTREP + 1
          GO TO 10
     END IF

***// CALCULATE THE AHAT SUB I'S (WEIGHT ESTIMATES)

     SUMNAI = 0.
     DO 165 I=1, K
          AHATI(I) = QHATI(I) / QHTMAX
          SUMNAI = SUMNAI + NN(I) * AHATI(I)
 165 CONTINUE
     IF(PRNT.EQ.1) THEN
          WRITE(1,0006)
          WRITE(1,C1C15)
          WRITE(1,3334) AHATI
     ELSE
     END IF
```

```
***// CALCULATE 1 REPLICATION OF UPPR ALFA C.L. ON SYSTEM RELIABILITY

      DEGFR(LOOP) = 2 * (1 + BIGF)

      DO 170 ALF=1, MAXALF
           CALL MDCHI(1 - ALFA(ALF),DEGFR(LOOP),CHISQ(ALF,LOOP), ERROR)
           QHTUPR(ALF,LOOP) = CHISQ(ALF,LOOP) / (2 * SUMNAI)
           IF(FLAG.EQ.1) THEN
                RHTSTR(ALF,LOOP) = 1 -(CHISQ(ALF,LOOP) / REAL(2*N(1)))
           ELSE
           END IF

***// CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR COMPNTS. IN SERIES

           CALL RHTSRS(QHTUPR(ALF,LOOP), AHATI,K, RSHAT(ALF,LOOP))
*    +T40,F8.5)') RSHAT(ALF,LOOP)
***// CALCULATE VALUE OF THE SYSTEM RELIABILITY FOR BRIDGE STRUCTURE ***

           CALL RHTBRG(QHTUPR(ALF,LOOP),AHATI,K,RSHTBR(ALF,LOOP))

 170  CONTINUE

***// THIS ELSE AND ENDIF ARE FOR THE TEST AGAINST MAXRUN             //***

      ELSE
           WRITE(1,'('' '',/''PROGRAM EXCEEDED THE MAX NO. OF RUNS'',
     +'' ALLOWED OF:'',I6)') TOTREP
      GOTO 9999

      END IF
      GOTO 10
      END IF

***// SORT THE ARRAYS OF SYSTEM UNRELIABILITIES(1 FOR EACH CONF. LEVEL)

      DO 700 ALF=1, MAXALF
           DO 800 REPS=1, MAXREP
                TRANSQ(REPS) = QHTUPR(ALF,REPS)
                TRANSR(REPS) = RSHAT(ALF,REPS)
                TRNSTR(REPS) = RHTSTR(ALF,REPS)
                TRANBR(REPS) = RSHTBR(ALF,REPS)
 800       CONTINUE
           CALL SHSORT(TRANSQ,KEY1,MAXREP)
           CALL SHSORT(TRANSR,KEY2,MAXREP)
           CALL SHSORT(TRNSTR,KEY3,MAXREP)
           CALL SHSORT(TRANBR,KEY4,MAXREP)
           DO 900 REPS=1, MAXREP
                QHTUPR(ALF,REPS) = TRANSQ(REPS)
                RSHAT(ALF,REPS) = TRANSR(REPS)
                RHTSTR(ALF,REPS) = TRNSTR(REPS)
                RSHTBR(ALF,REPS) = TRANBR(REPS)
 900       CONTINUE
 700  CONTINUE
```

```
***// PRINT OUTPUT REPORT HEADINGS                                //***
      WRITE(1,6666)
      WRITE(1,6667) MAXREP
      WRITE(1,6668) K
      WRITE(1,6770) SCALEN
      WRITE(1,6669)
      IF(K.EQ.5) THEN
           WRITE(1,6699)
      ELSE
      END IF
      WRITE(1,6670) MSTRQ
      WRITE(1,6671)
      WRITE(1,C1C15)
      WRITE(1,3334) AI
      WRITE(1,0007)
      WRITE(1,C1C15)
      WRITE(1,3334) QI
      WRITE(1,0005)
      WRITE(1,C1C15)
      WRITE(1,3335) N
      WRITE(1,6674)
      WRITE(2,6666)
      WRITE(2,6667) MAXREP
      WRITE(2,6668) K
      WRITE(2,6770) SCALEN
      WRITE(2,6669)
      IF(K.EQ.5) THEN
           WRITE(1,6699)
      ELSE
      END IF
      WRITE(2,6670) MSTRQ
      WRITE(2,6671)
      WRITE(2,C1C15)
      WRITE(2,3334) AI
      WRITE(2,0007)
      WRITE(2,C1C15)
      WRITE(2,3334) QI
      WRITE(2,0005)
      WRITE(2,C1C15)
      WRITE(2,3335) N
      WRITE(2,6674)
      WRITE(2,'(''SORTED RSHAT 1  IS:'',/15(F8.5))')
     +(RSHAT(1,REPS), REPS=1, MAXREP)
      WRITE(2,'(''SORTED RSHAT 2  IS:'',/15(F8.5))')
     +(RSHAT(2,REPS), REPS=1, MAXREP)
      IF(FLAG.EQ.1) THEN
           WRITE(2,'(''SORTED RHTSTR 1  IS:'',/10(F8.5))')
     +(RHTSTR(1,REPS), REPS=1, MAXREP)
           WRITE(2,'(''SORTED RHTSTR 2  IS:'',/10(F8.5))')
     +(RHTSTR(2,REPS), REPS=1, MAXREP)
      ELSE
      END IF
      IF(K.EQ.5) THEN
           WRITE(2,'(''SORTED RSHTBR 1  IS:'',/10(F8.5))')
     +(RSHTBR(1,REPS), REPS=1, MAXREP)
```

```
        WRITE(2,'('' SORTED RSHTBR 2  IS:'',/10(F8.5))')
       +(RSHTBR(2,REPS), REPS=1, MAXREP)
        ELSE
        END IF

***// COMPUTE THE VALUE RS OF THE TRUE SYSTEM REL. FNCTN.(SERIES SYSTEM)
***// AND FOR THE 5-COMPONENT BRIDGE STRUCTURE

        CALL RSRS(QI,K,RS)
        WRITE(1,'('' '',////''THE TRUE SERIES SYSTEM '',
       +''RELIABILITY VALUE IS:'',T51,F8.5)') RS
        CALL RBRIDG(QI,K,RSBRDG)
        IF(K.EQ.5) THEN
        WRITE(1,'('' '',////''THE TRUE BRIDGE STRUCTURE '',
       +''RELIABILITY VALUE IS:'',T51,F8.5)') RSBRDG
        ELSE
        END IF
        WRITE(1,6675)

***// COMPUTE THE DIFFERENCE 'DELTAR' BTWN. RS AND RSHAT OF THE THEO
***// RETICAL QUANTILE GIVEN BY ALFA(MUST USE SORTED RSHAT ARRAY)

        IF(FLAG.EQ.1) THEN
             WRITE(1,5755)
        ELSE
        END IF
        DO 450 ALF=1, MAXALF
             QUANTL(ALF) = MAXREP * (1 - ALFA(ALF))
             DELTAR(ALF) = RS - RSHAT(ALF,QUANTL(ALF))
             DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))
             IF(FLAG.EQ.1) THEN
                  DELSTR(ALF) = RS - RHTSTR(ALF,QUANTL(ALF))
                  WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
                  WRITE(1,5656) RHTSTR(ALF,QUANTL(ALF))
                  WRITE(1,5657) DELSTR(ALF)
             ELSE
             END IF
             IF(K.EQ.5) THEN
                  DELBRG(ALF) = RSBRDG - RSHTBR(ALF,QUANTL(ALF))
                  WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
                  WRITE(1,5666) RSHTBR(ALF,QUANTL(ALF))
                  WRITE(1,5667) DELBRG(ALF)
             ELSE
             END IF
             WRITE(1,5555) MAXREP, ALFA(ALF), REAL(QUANTL(ALF))
             WRITE(1,5556) RSHAT(ALF,QUANTL(ALF))
             WRITE(1,5557) DELTAR(ALF)
  450   CONTINUE

***// FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE //***
***//                 ********* RSHAT ********                    //***
        WRITE(1,6676)
        DO 400 ALF=1,MAXALF
        TRUQNT(ALF) = 0
             DO 500 REPS=1, MAXREP
                  DIFF(REPS) = RS - RSHAT(ALF,REPS)
```

```
500       CONTINUE
          DO 600 REPS=1, MAXREP
               IF(ABS(DIFF(REPS)).LE.EPS) THEN
                    TRUQNT(ALF) = REPS
                    WRITE(1,'('' '',/''TRUE CONFIDENCE LIMIT IS:'',
     +              F8.4)')
     +              (TRUQNT(ALF) / REAL(MAXREP)) * 100.
                    GO TO 620
               ELSEIF(DIFF(REPS).LT.0.) THEN
                    TRUQNT(ALF) = REPS
                    GO TO 610
               ELSE
               END IF
600       CONTINUE
610       IF(TRUQNT(ALF).EQ.0.) THEN
               WRITE(1,4443) ALFA(ALF)
               WRITE(1,'('' '',/''THE SMALLEST'',
     +         '' DIFFERENCE BETWEEN RS AND RSHAT IS:'',F10.5)') DIFF(
     +         MAXREP)
          ELSEIF(TRUQNT(ALF).EQ.1.) THEN
               WRITE(1,4442) ALFA(ALF)
               WRITE(1,'('' '',/''ALL RSHAT'',
     +          '' ARE GREATER THAN RS'')')
          ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))
     +    THEN
               WRITE(1,4444) ALFA(ALF),
     +         (TRUQNT(ALF) / REAL(MAXREP)) * 100.
               WRITE(1,4445) RSHAT(ALF,TRUQNT(ALF))
               WRITE(1,4446)
          ELSE
               WRITE(1,4444) ALFA(ALF),
     +         ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
               WRITE(1,4445) RSHAT(ALF,TRUQNT(ALF)-1)
               WRITE(1,4447)
620       END IF
400  CONTINUE


***// FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE //***
***//        ********* RSHTBR (BRIDGE) *******                    //***

     IF(K.EQ.5) THEN
     DO 401 ALF=1,MAXALF
     TRUQNT(ALF) = 0
          DO 501 REPS=1, MAXREP
               DIFF(REPS) = RSBRDG - RSHTBR(ALF,REPS)
501       CONTINUE
          DO 601 REPS=1, MAXREP
               IF(ABS(DIFF(REPS)).LE.EPS) THEN
                    TRUQNT(ALF) = REPS
                    WRITE(1,'('' '',/''TRUE CONFIDENCE LIMIT IS:'',
     +              F8.4)')
     +              (TRUQNT(ALF) / REAL(MAXREP)) * 100.
                    GO TO 621
               ELSEIF(DIFF(REPS).LT.0.) THEN
                    TRUQNT(ALF) = REPS
```

```
                    GO TO 611
                ELSE
                END IF
601         CONTINUE
611         IF(TRUQNT(ALF).EQ.0.) THEN
                WRITE(1,4443) ALFA(ALF)
                WRITE(1,'('' '',/''THE SMALLEST'',
     +          '' DIFFERENCE BETWEEN RSBRDG AND RSHTBR IS:'',
     +          F10.5)') DIFF(MAXREP)
            ELSEIF(TRUQNT(ALF).EQ.1.) THEN
                WRITE(1,4442) ALFA(ALF)
                WRITE(1,'('' '',/''ALL RSHTBR'',
     +           '' ARE GREATER THAN RSBRDG'')')
            ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))
     +      THEN
                WRITE(1,4444) ALFA(ALF),
     +          (TRUQNT(ALF) / REAL(MAXREP)) * 100.
                WRITE(1,4449) RSHTBR(ALF,TRUQNT(ALF))
                WRITE(1,4446)
            ELSE
                WRITE(1,4444) ALFA(ALF),
     +          ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
                WRITE(1,4449) RSHTBR(ALF,TRUQNT(ALF)-1)
                WRITE(1,4447)
621         END IF
401     CONTINUE
        ELSE
        END IF


***// FIND THE TRUE CONFIDENCE LEVEL OF THE SYSTEM REL. ESTIMATE //***
***//                     ********* RHTSTR *********                 //***

        IF(FLAG.EQ.1) THEN
        DO 4400 ALF=1,MAXALF
        TRUQNT(ALF) = 0
            DO 5500 REPS=1, MAXREP
                DIFF(REPS) = RS - RHTSTR(ALF,REPS)
5500        CONTINUE
            DO 6600 REPS=1, MAXREP
                IF(ABS(DIFF(REPS)).LE.EPS) THEN
                    TRUQNT(ALF) = REPS
                    WRITE(1,'('' '',/''TRUE CONFIDENCE LIMIT IS:'',
     +              F8.4)')
     +              (TRUQNT(ALF) / REAL(MAXREP)) * 100.
                    GO TO 6620
                ELSEIF(DIFF(REPS).LT.0.) THEN
                    TRUQNT(ALF) = REPS
                    GO TO 6610
                ELSE
                END IF
6600        CONTINUE
6610        IF(TRUQNT(ALF).EQ.0.) THEN
                WRITE(1,4443) ALFA(ALF)
                WRITE(1,'('' '',/''THE SMALLEST'',
     +          '' DIFFERENCE BETWEEN RS AND RHTSTR IS:'',
```

```
     +         F9.5)') DIFF(MAXREP)
          ELSEIF(TRUQNT(ALF).EQ.1.) THEN
               WRITE(1,4442) ALFA(ALF)
               WRITE(1,'('' '',/''ALL RHTSTR'',
     +         '' ARE GREATER THAN RS'')')
          ELSEIF(ABS(DIFF(TRUQNT(ALF))).LE.ABS(DIFF(TRUQNT(ALF) - 1)))
     +    THEN
               WRITE(1,4444) ALFA(ALF),
     +         (TRUQNT(ALF) / REAL(MAXREP)) * 100.
               WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF))
               WRITE(1,4446)
          ELSE
               WRITE(1,4444) ALFA(ALF),
     +         ((TRUQNT(ALF)-1) / REAL(MAXREP)) * 100.
               WRITE(1,4448) RHTSTR(ALF,TRUQNT(ALF)-1)
               WRITE(1,4447)
 6620     END IF
 4400 CONTINUE
      ELSE
      END IF

***// PRINT THE ARRAYS PERTINENT TO THE OUPUT OF EACH REPLICATION //***

      IF(PRNT.EQ.1) THEN
      I = 1
 185  WRITE(1,REPSHD) ALFA(SELCTA), ALFA(SELCTA),
     +ALFA(SELCTB),ALFA(SELCTB),ALFA(SELCTA),ALFA(SELCTA),ALFA(SELCTB),
     +ALFA(SELCTB)
 175  IF(I.GE.(MAXREP + 1)) THEN
          GOTO 180
      ELSE
          IF( (I.EQ.71).OR.(I.EQ.211).OR.(I.EQ.351).OR.(I.EQ.491).OR.
     +    (I.EQ.631).OR.(I.EQ.771).OR.(I.EQ.911).OR.(I.EQ.1051) ) THEN
               I = I + 70
               WRITE(1,'(''+'')')
               GOTO 185
          ELSE

          WRITE(1,3336) I, INT(DEGFR(I)), CHISQ(1,I), QHTUPR(1,I),
     +    CHISQ(2,I), QHTUPR(2,I)
          END IF
          IF(I + 70.LE.MAXREP) THEN
               WRITE(1,3337) I+70,INT(DEGFR(I+70)),CHISQ(1,I+70),
     +         QHTUPR(1,I+70),CHISQ(2,I+70),QHTUPR(2,I+70)
          ELSE
          END IF
      I = I + 1
      GOTO 175
 180  END IF
      ELSE
      ENDIF
 9999 WRITE(1,'(''THE TOTAL NO OF REPS WAS:'',I8)') TOTREP
      WRITE(1,'(''THE TOTAL NO OF EFFECTIVE REPS WAS:'',I8)') LOOP
      WRITE(1,'(''THE TOTAL NO OF NO FAILURE RUNS WAS:'',I8)') ZFAILS
      WRITE(1,'(''THE TOTAL NO OF RUNS WITH FAILURES WAS:'',I8)') FAILS
      WRITE(1,'(''AVG NO OF COMPONENTS WITH NO FAILURES PER REP WAS:'',
```

```
    +F5.2)') ZFPREP / MAXREP
0008 FORMAT (/ 3X,'C 1',5X,'C 2',
    +5X,'C 3',5X,'C 4',5X,'C 5',5X,'C 6',5X,'C 7',5X,
    +'C 8',5X,'C 9',5X,'C 10',4X,'C 11',4X,
    +'C 12',4X,'C 13',4X,'C 14',4X,'C 15')
0009 FORMAT(/1X,'REP NO',2X,'DF',1X,'CHISQR(',F4.3,')',1X,
    +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'QHTUPR(',F4.3,')',
    +2X,'REP NO',2X,'DF',1X,'CHISQR(',F4.3,')',1X,
    +'QHTUPR(',F4.3,')',1X,'CHISQR(',F4.3,')',1X,'QHTUPR(',F4.3,')'/)
0001 FORMAT (///'UNIFORM RANDOM DEVIATES ARE: ')
0002 FORMAT (///'BERNOULLI TRIALS ARE:')
0003 FORMAT (///'TOTAL NO. OF FAILURES FOR EACH COMPONENT:')
0004 FORMAT (///'ESTIMATED UNRELIABILITY FOR EACH COMPONENT:')
0005 FORMAT (///'TOTAL NUMBER OF MISSION TESTS:')
0006 FORMAT (///'ESTIMATED WEIGHTS FOR EACH COMPONENT:')
0007 FORMAT (///'Q I FOR EACH COMPONENT:')
1111 FORMAT (15F8.5)

2222 FORMAT (/1X,15(I4,4X))
3333 FORMAT (/1X,15(I4,4X))
3334 FORMAT (/15F8.5)
3335 FORMAT (/1X,15(I4,4X))
3336 FORMAT (T3,I4,T9,I3,T13,F11.5,T27,F8.5,T39,F11.5,T53,F8.5)
3337 FORMAT ('+',T67,I4,T73,I3,T77,F11.5,T91,F8.5,T103,F11.5,T117,F8.5)
4442 FORMAT (' ',///'THE RESULTING (1 - ',F4.3,') CONFIDENCE ',
    +'LIMIT IS:',T50,' 00.000 ')
4443 FORMAT (' ',///'THE RESULTING (1 - ',F4.3,') CONFIDENCE ',
    +'LIMIT IS:',T50,'100.0000')
4444 FORMAT (' ',///'THE RESULTING (1 - ',F4.3,') CONFIDENCE ',
    +'LIMIT IS:',T50,F8.4)
4445 FORMAT (' ',/'THE RSHAT VALUE CLOSEST TO RS IS: ',T51,F8.5)
4446 FORMAT (' ',/'(FIRST NEGATIVE DIFFERENCE)')
4447 FORMAT (' ',/'(ELEMENT PRECEEDING FIRST NEGATIVE DIFFERENCE)')
4448 FORMAT (' ',/'THE RHTSTR VALUE CLOSEST TO RS IS: ',T51,F8.5)
4449 FORMAT (' ',/'THE RSHTBR VALUE CLOSEST TO RSBRDG IS: ',T51,F8.5)
5555 FORMAT (' ',///'THE ',I4,'(1-',F4.3,') QUANTILE IS:',T49,F8.3)
5556 FORMAT (' ',/'THE VALUE OF RSHAT FOR THAT QUANTILE IS:',T51,F8.5)
5557 FORMAT (' ',/'THE DIFFERENCE(RS - RSHAT) IS:',T51,F8.5)
5656 FORMAT (' ',/'THE VALUE OF RHTSTR FOR THAT QUANTILE IS:',T51,F8.5)
5666 FORMAT (' ',/'THE VALUE OF RSHTBR FOR THAT QUANTILE IS:',T51,F8.5)
5657 FORMAT (' ',/'THE DIFFERENCE(RS - RHTSTR) IS:',T51,F8.5)
5667 FORMAT (' ',/'THE DIFFERENCE(RS - RSHTBR) IS:',T51,F8.5)
5755 FORMAT (' ',///'SINCE THE NO. OF MISSION TESTS IS THE SAME FOR',
    +' ALL COMPONENTS THE CLOSED FORM SERIES SYSTEM RELIABILITY ',
    +'''RHTSTR'' IS COMPUTED')
6666 FORMAT ('+','************************************************',
    +'* RUN INPUT SETTINGS FOR ADJUSTED SUM OF THE NI AI ************',
    +'************************')
6667 FORMAT (' ',//'NUMBER OF REPLICATIONS:',T50,I4)
6668 FORMAT (' ',//'NUMBER OF COMPONENTS:',T50,I4)
6669 FORMAT (' ',//'SYSTEM RELIABILITY FUNCTION:',T50,'SERIES')
6699 FORMAT (' ',//'SYSTEM RELIABILITY FUNCTION:',T50,'BRIDGE')
6670 FORMAT (' ',//'MASTER UNRELIABILITY USED:',T50,F8.5)
6770 FORMAT (' ',//'ZERO FAILURE SCALING FACTOR:',T50,F5.2)
6671 FORMAT (' ',//'INPUT WEIGHTS(A SUB I''S):')
6674 FORMAT ('+',///'*************************************************',
```

```
     +'**R U N  R E S U L T S****************************************',
     +'**************')
6675 FORMAT ('+',///'************************************************',
     +' ESTIMATE ERRORS **********************************************',
     +'***************')
6676 FORMAT ('+',///'************************************************',
     +' TRUE CONFIDENCE LIMITS ***************************************',
     +'***************')
      END
```

APPENDIX D. EXTERNAL SUBROUTINES

These six subroutines are used by all three programs listed in appendices one through three. They must be appended to the particular main program being run or they must be available on the same disk as the one from which the main program is being run.

```
      SUBROUTINE IMAX(SEQ, N, MX, INDEX)

***// THIS ROUTINE IDENTIFIES THE MAXIMUM ELEMENT OF AN INTEGER VECTOR
*     NOTE THAT SINCE THE TEST IS .GT. SUBROUTINE ONLY PICKS THE FIRST
*          OCCURRENCE OF A MAX SUCH AS IN THE CASE OF A TIE.
***//      ALSO, ARRAY PASSED MUST BE TYPE INTEGER
      REAL*4 MX
      INTEGER SEQ(N), N, I, INDEX
      MX = 0.
      DO 5 I=1, N
           IF(SEQ(I).GT.MX) THEN
               MX = SEQ(I)
               INDEX = I
           ELSE
           END IF
 5    CONTINUE
      END


      SUBROUTINE RBRIDG(QI,N,RRSS)

***// THIS SUBROUTINE CALCULATES THE "TRUE" RELIABILITY OF A 5-COMPONENT
***// BRIDGE STRUCTURE
      REAL*4 QI(N), RRSS
      INTEGER N
      IF(N.NE.5) THEN
           WRITE(1,'(''WARNING: BRIDGE STRUCTURE ONLY USES '',
     +''THE FIRST 5 COMPONENTS'')')
      ELSE
      END IF
      RRSS=(1-QI(1))*(1-QI(4))+(1-QI(2))*(1-QI(5))+(1-QI(1))*(1-QI(3))*
     C(1-QI(5))+(1-QI(2))*(1-QI(3))*(1-QI(4))-(1-QI(1))*(1-QI(2))*
     C(1-QI(3))*(1-QI(4))-(1-QI(1))*(1-QI(2))*(1-QI(3))*(1-QI(5))-
     C(1-QI(1))*(1-QI(2))*(1-QI(4))*(1-QI(5))-(1-QI(1))*(1-QI(3))*
     C(1-QI(4))*(1-QI(5))-(1-QI(2))*(1-QI(3))*(1-QI(4))*(1-QI(5))+
     C2*(1-QI(1))*(1-QI(2))*(1-QI(3))*(1-QI(4))*(1-QI(5))
      END


      SUBROUTINE RHTBRG(QHTUP,AHT,N,RRBRDG)
***// THIS SUBROUTINE CALCULATES THE ESTIMATED RELIABILITY OF A
***// 5-COMPONENT BRIDGE STRUCTURE.(ONLY CARRIED OUT TO THE Q-CUBED TERM
      REAL*4 QHTUP, RRBRDG, AHT(N)
      INTEGER N
      RRBRDG=1-((QHTUP**2)*(AHT(1)*AHT(2)+AHT(4)*AHT(5)))-
     C((QHTUP**3)*(AHT(1)*AHT(3)*AHT(5)+AHT(2)*AHT(3)*AHT(4)))+
```

```
     C((QHTUP**4)*(AHT(1)*AHT(2)*AHT(3)*AHT(4)+AHT(1)*AHT(2)*AHT(3)*
     CAHT(5)+AHT(1)*AHT(2)*AHT(4)*AHT(5)+AHT(1)*AHT(3)*AHT(4)*AHT(5)+
     CAHT(2)*AHT(3)*AHT(4)*AHT(5)))-
     C2*((QHTUP**5)*(AHT(1)*AHT(2)*AHT(3)*AHT(4)*AHT(5)))
      END

      SUBROUTINE RHTSRS(QHTUP,AAHTI,N,RRSHAT)

***// THIS ROUTINE CALCULATES THE VALUE OF THE SYSTEM RELIABILITY OF A
*     SERIES SYSTEM OF 'N' NO. OF COMPONENTS WHICH HAVE UNRELIABILITY
***// 'QHTUP'.  THE FINAL SYSTEM RELIABILITY VALUE PASSED IS 'RRSHAT'
      REAL*4 QHTUP, RRSHAT, AAHTI(N)
      INTEGER I, N
      RRSHAT = 1.
      DO 100 I=1, N
           RRSHAT = RRSHAT * (1 -(AAHTI(I)* QHTUP))
 100  CONTINUE
      END

      SUBROUTINE RMAX(SEQ, N, MX, INDEX)
***// THID ROUTINE IDENTIFIES THE MAXIMUM ELEMENT OF A REAL VECTOR
*     NOTE THAT SINCE THE TEST IS .GT. SUBROUTINE ONLY PICKS THE FIRST
*          OCCURRENCE OF A MAX SUCH AS IN THE CASE OF A TIE.
***//      ALSO, ARRAY PASSED MUST BE TYPE REAL
      REAL*4 MX, SEQ(N)
      INTEGER N, I, INDEX
      MX = 0.
      DO 5 I=1, N
           IF(SEQ(I).GT.MX) THEN
               MX = SEQ(I)
               INDEX = I
           ELSE
           END IF
 5    CONTINUE
      END

      SUBROUTINE RSRS(QIS,N,RRS)

***// THIS ROUTINE CALCULATES THE VALUE OF THE SYSTEM RELIABILITY OF A
*     SERIES SYSTEM OF 'n' COMPONENTS WHICH HAVE UNRELIABILITY
***// 'QIS'.  THE FINAL SYSTEM RELIABILITY VALUE PASSED IS 'RRS'
      REAL*4 QIS(N), RRS
      INTEGER I, N
      RRS = 1.
      DO 100 I=1, N
           RRS = RRS * (1 - QIS(I))
 100  CONTINUE
      END
```

LIST OF REFERENCES

1. Myhre, J. M. and S. C. Saunders, *Comparison of Two Methods of Obtaining Approximate Confidence Intervals for System Reliability*, Technometrics, 10, pp.37-49, 1968.

2. Rao, C. R., *Advanced Statistical Methods in Biometric Research* , John Wiley and Sons, Inc., New York, 1952.

3. Easterling, Robert G., *Approximate Confidence Limits for System Reliability*, Journal of the American Statistical Association, 67, pp.220-2, March 1972.

4. Winterbottom, Alan, *Lower Confidence Limits for Series System Reliability from Binomial Subsystem Data*, Journal of the American Statistical Association, 69, pp.782-787, September 1974.

5. Mann, Nancy R. and Frank E. Grubbs., *Approximatly Optimum Confidence Bounds for System Reliability Based on Component Test Data.*, Technometrics, 16, pp.335-47, 1974.

6. Mann, Nancy R., Ray E. Schafer and Nozer D. Singpurwalla, *Methods for Statistical Analysis of Reliability and Life Data.*, John Wiley and Sons, Inc. New York, pp.487-524, 1974.

7. Woods, W. M. and J. R. Borsting, *A Method for Computing Confidence Limits on System Reliability Using Component Failure Data With Unequal Sample Sizes*, Naval Postgraduate School Technical Report, June 1968.

8. Lomnicki, Z. A., *Two Terminal Series-Parallel Networks*, Advances in Applied Probability, 4, pp.109-150, 1973.

9. Myhre, J., Rosenfeld, A., and Saunders, S. *Determining Confidence Bounds for Highly Reliable Coherent Systems Based on a Paucity of Component Failures,* Naval Research Logistics Quarterly, 25, pp.213-227, June 1978.

10. Lee, Hyeon-Soo, *Approximate Interval Estimation Methods for the Reliability of Systems Using Component Data With Exponential and Weibull Distributions,* M.S. Thesis, Naval Postgraduate School, Monterey, California, September 1989.

11. Lloyd, D. and Lipow, M., *Reliability: Management, Methods, and Mathematics,* 2nd Edition, American Society for Quality Control, 1984.

BIBLIOGRAPHY

Thesis Manual, Naval Postgraduate School, Monterey, California, 1988.

Efron, Bradley and Gail Gong, *A Leisurely Look at the Bootstrap, the Jacknife, and Cross-Validation,* The American Statistician, 37, No. 1, February 1983.

Efron, Bradley, *Nonparametric Estimates of Standard Error: The Jacknife, the Bootstrap and other Methods,* Biometrika, 68, 3, pp.589-99, 1981.

Nyhoff, L. and S. Leestma, *FORTRAN 77 for Engineers and Scientists,* Macmillan Publishing Company, New York, 1985.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Professor W. Max Woods Naval Postgraduate School, Code 55Wo Monterey, CA 93943-5000	3
4.	Professor Harold J. Larson Naval Postgraduate School, Code 55La Monterey, CA 93943-5000	1
5.	Chief of Naval Operations(OP-81) Navy Department Washington, DC 20350	1
6.	Commander, Pacific Missile Test Center Attn: Glatha Hemme Point Mugu, CA 93042	1
7.	Gary Walraven 1555 Rockridge Court Colorado Springs, CO 80918	1
8.	Dinorah R. Bellini 8881B Fountainbleu Blvd., B-207 Miami FL, 33172	1
9.	Value Advice, Inc. 8660 N.W. 6th Lane Suite 208 Miami, FL 33126	1
10.	Lt. Edmundo F. Bellini 1368 Marshall Lane Virginia Beach, VA 23455	2