COMPUTERISED DATA RETRIEVAL:
AN EXPERIMENT WITH IBM 1620*

L J HARAVU
Indian Institute of Technology, New Delhi.

A S RAIZADA
Insdoc, New Delhi—12.

The objective of this experiment was to find out the suitability in the storage and retrieval of data using the IBM 1620 computer facilities and the Fortran IIID language. The IAEA's Directory of Nuclear Reactors, vol 1—Power Reactors, was used as the data source for the experiment. The method followed in encoding and programming are described. The results of programmes for search and listing are presented.

INTRODUCTION

The experimental cell of the Projects Division, having gained familiarity with problems like computer processing of the Union Catalogue data and preparation of indexes by computer, devoted some thought to problems in data retrieval with the facilities available to the cell. The opportunity was provided when one of the authors was deputed for the IIInd Training Course in Documentation and Reprography, at Insdoc during the year 1965-66. As part of this Training Course a data retrieval project was assigned and had to be completed during the period of the course. The present article reports the work done during this period.

This experiment on data retrieval is based upon the data available in the IAEA Directory of Nuclear Reactors Vol. I, Power Reactors, referred to here as Data Source. The accent in this experiment has not been on the collection of data from primary sources. Rather, the attempt has been one of using this as source for experiment with the objectives delineated below:

1. Familiarization with the IBM 1620 computer system;
2. Gaining experience in storing data on disk;
3. Retrieving data through the computer; and
4. Producing intelligible lists of data.

The Data Source

The purpose of the data source is to make available important details of the various power reactor projects. Information is presented in such a way as to provide easy reference to technical staff and management in the field of atomic physics. Information is presented in an uniform way for each reactor. The information for each reactor is presented under major heads such as 'general information', 'reactor physics', 'core data' etc. Each of these major heads is broken down further into subheads (e.g. under the major head core we find subheads—shape and dimensions, No. of channels and subassemblies, lattice, critical mass etc.) The qualitative and quantitative information pertains to specific heads under each of these subheads (e.g. under the subhead dimensions, data on length (specific head), breadth etc. may be provided). No attempt in this experiment, has been made to add, verify, modify or supplement the data already available in the data source.

Method

The work in this experiment may be broadly divided into: (1) Encoding; and (2) Programming. In this part are considered method and details in encoding and programming.

Encoding

The word 'encoding' as used here includes all the operations necessary to record the data source onto punched cards. The operations involved in order of the flow of work were:

1. Decision as to items to be recorded;
2. Field allocation for punched cards;
3. Transcription of data onto code sheets; and
4. Key punching of data onto cards, and verification.
Decision as to items to be recorded

An item of data is characterized by three variables: the kind of measurement (length, breadth, gross electricity etc.), the numerical value of the measurement, and the units of measurement. It was necessary to decide whether to record all, some or any combination of these variables. It was decided that only the numerical value of a measurement and the units of measurement would be recorded. At the time of search and listing the kind of measurement searched or listed would be taken care of by programming. It was also decided not to include items that were narrative in nature.

Specific heads

Another decision taken was to present the data under the same major heads and subheads as found in the data source (See item 6 in the Appendix). The specific heads under each subhead posed a problem. In order to maintain uniformity, it was necessary to have the same specific heads for each reactor under each subhead. This involved the scanning of the directory under each subhead to call out the relevant specific heads occurring under each subhead. This has the disadvantage of providing certain specific heads which were not relevant to an item of data. (e.g. the dimensions of a cylinder can be fully represented by the height, and diameter. Here length and breadth would be superfluous). The provision of uniform specific heads for each reactor, however, has two advantages: (1) ease in allocation of fields on punched cards; (2) provision of space on punched cards for data that is not available now, but may become available subsequently.

An example of the specific heads provided by this experiment under the major head 'core' is presented below. Words occurring at the first indentation are the subheads and those at the second indentation are the specific heads.

MAJOR HEAD: CORE

SHAPE
DIMENSIONS
  DIAMETER/BREADTH
  HEIGHT
  LENGTH
NO. OF CHANNELS
  FUEL
  CONTROL ROD
  FUEL ELEMENT
  SAFETY ROD

NO. OF SUBASSEMBLIES
  FUEL ROD
  STATIONARY
  MOVEABLE
LATTICE
  SHAPE
  PITCH
  FUEL RODS
  SUBASSEMBLIES
SPACING
CRITICAL MASS
CORE LOADING AT RATED POWER
AVG. SPECIFIC POWER IN FUEL
AVG. POWER DENSITY IN CORE
  SEED
  BLANKET
BURN UP
  AVG.
  MAV.
  PERCENTAGE
REFUELLING SCHEDULE
  PARTIAL
  FULL
DOWN TIME
MODERATOR
WEIGHT/QUANTITY
  IN CORE
  AVG. TEMP
  MAX. TEMP
SEED
  Blanket
  AVG.
  MAV.
  Percentage
REFUELLING SCHEDULE
  PARTIAL
  FULL
DOWN TIME
MODERATOR
WEIGHT/QUANTITY
  IN CORE
  AVG. TEMP
  MAX. TEMP

Such decisions were taken for all the major heads.

Allocation of fields

Field allocation in the experiment was based on empirical considerations and proceeded along the following lines:

1. The data source was scanned at random under each subhead and specific head to find out the maximum number of columns required for data under any specific head of information. For instance, if say, the length (numerical value plus units of measurement) of an item did not exceed 6 characters, 8 to 10 columns were provided on the card. This was with a view to accommodate exceptional cases that might occur.

2. The field length, thus selected was put to test and any changes in the field length necessary was made.
3. The number of cards required to record the data for each reactor under any major head of information was more than one in all cases. It was, therefore, necessary to allocate one field on each card for the purposes of identification and arrangement, in case a deck of data cards got disarranged. The field chosen for this purpose was columns 74 to 80. The columns 74 and 75 were devoted to the card number. Columns 76, 77, 78 were devoted to the reactor serial number. Columns 79 and 80 were devoted to the major head (General Information, Core etc.). Thus for instance, if the punching in these columns was 0100103 it meant the first card or 01 card, of reactor with serial number 001, of the major head 'core' (core being given the code 03).

With the help of this field, it was possible to maintain all cards pertaining to data under any major head of any reactor in strict order. This is important since any inaccuracy in the order in which data is fed into the computer will result in inaccurate lists as well as wrong search results.

Given in the next page is an example of the field allocation done for the specific heads of data under the major head 'core'.

**Transcription of data to code sheets**

The recording of data on code sheets was done in strict accordance with the field allocation. Thus for instance if on card 01 of the major head core columns 1-26 were allocated for core shape, the shape (e.g. cylinder, polygonal prism) was recorded, as such in these columns. The next item of data was recorded from column 29 onwards and 50 on. An example of data recorded on a code sheet is presented in appendix 15).

**Conversion**

The data on code sheets were now punched and verified using the IBM 024 and 056 punch and verifier respectively. The card file was now ready for further processing on the computer.

**Programming**

The programming done in this experiment may be considered under three heads:

1. Programmes for the storage of data;
2. Programmes for the retrieval of data as answers to queries; and
3. Programmes for the listing of data in an intelligible form.

**Programming for the Storage of data:**

The storage of large masses of data constitutes one of the basic operations in a data retrieval system. The basic operation in this experiment was to store data on disks, to facilitate further manipulation either for retrieval or for listing.

The data that was encoded on punched cards, had to be first read into the internal memory before being transferred to disk. The method of loading the data into the memory was as follows:

Items of data under specific heads of a major head were made elements of a matrix. Each column in this matrix represented the data under a major head for one reactor. Thus for instance, core data for all the reactors would internally be represented as the matrix below (The figures in the parentheses indicate the row and column number).

| CORE (1,1) | CORE (1,2) | ... | CORE (1,n) |
| CORE (2,1) | CORE (2,2) | ... | CORE (2,n) |
| CORE (3,1) | CORE (3,2) | ... | CORE (3,n) |
|            |            | ... |            |
| CORE (m,1) | CORE (m,2) | ... | CORE (m,n) |

'Core' is the name given to the matrix by which data on the reactor core would be referred to internally in the computer. The quantity 'm' represents the number of elements in core data and 'n' represents the number of reactors. The value of m was different for different major heads, while n remained constant. The values m, n together constitute the dimension of a matrix.

Each element or a consecutive group of elements of a column of this matrix represents data under any specific head. For instance elements CORE (1,1); CORE (2,1); and CORE (3,1) together would internally represent the data under the specific head 'shape of core' for the reactor number 1. Generalizing CORE (1,n); CORE (2,n); and CORE (3,n) together would internally represent the core shape of reactor number 'n'.
## COMPUTERISED DATA RETRIEVAL

### MAJOR HEAD: CORE (CODE : 03 in Cols 79 and 80)

<table>
<thead>
<tr>
<th>SUBHEAD</th>
<th>SPECIFIC HEAD</th>
<th>CARD COLS</th>
<th>CARD NO. (CODE IN COLS. 74-75)</th>
<th>LENGTH OF FIELD</th>
</tr>
</thead>
<tbody>
<tr>
<td>SHAPE</td>
<td></td>
<td>1-28</td>
<td>01</td>
<td>28</td>
</tr>
<tr>
<td>DIMENSIONS</td>
<td>DIAMETER/BREADTH</td>
<td>29-36</td>
<td>01</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>HEIGHT</td>
<td>37-44</td>
<td>01</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>LENGTH</td>
<td>45-52</td>
<td>01</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>FUEL</td>
<td>53-58</td>
<td>01</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>CONTROL ROD</td>
<td>59-64</td>
<td>01</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>FUEL ELEMENT</td>
<td>65-70</td>
<td>01</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>SAFETY ROD</td>
<td>7-12</td>
<td>02</td>
<td>6</td>
</tr>
<tr>
<td>NO. OF CHANNELS</td>
<td></td>
<td>13-18</td>
<td>02</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>FUEL ROD</td>
<td>19-24</td>
<td>02</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>STATIONARY</td>
<td>25-30</td>
<td>02</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>MOVEABLE</td>
<td>31-36</td>
<td>02</td>
<td>6</td>
</tr>
<tr>
<td>LATTICE</td>
<td>SHAPE</td>
<td>37-54</td>
<td>02</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>PITCH</td>
<td>55-62</td>
<td>02</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>FUEL RODS</td>
<td>63-70</td>
<td>02</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>SUBASSEMBLIES</td>
<td>1-8</td>
<td>03</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>SPACING</td>
<td>9-14</td>
<td>03</td>
<td>6</td>
</tr>
<tr>
<td>CRITICAL MASS</td>
<td>AT (TEMPERATURE)</td>
<td>15-28</td>
<td>03</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td></td>
<td>29-37</td>
<td>03</td>
<td>9</td>
</tr>
<tr>
<td>CORE LOADING AT RATED POWER</td>
<td></td>
<td>38-65</td>
<td>03</td>
<td>28</td>
</tr>
<tr>
<td>AVG. SPECIFIC POWER IN FUEL</td>
<td></td>
<td>1-28</td>
<td>04</td>
<td>28</td>
</tr>
<tr>
<td>AVG. POWER DENSITY IN CORE</td>
<td></td>
<td>29-42</td>
<td>04</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>SEED</td>
<td>43-54</td>
<td>04</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>BLANKET</td>
<td>55-66</td>
<td>04</td>
<td>12</td>
</tr>
<tr>
<td>BURN UP</td>
<td>AVG.</td>
<td>1-14</td>
<td>05</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>MAX.</td>
<td>15-28</td>
<td>05</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>PERCENTAGE</td>
<td>29-30</td>
<td>05</td>
<td>2</td>
</tr>
<tr>
<td>REFUELLING SCHEDULE</td>
<td></td>
<td>31-33</td>
<td>05</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>FULL</td>
<td>34-36</td>
<td>05</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>TIME INTERVAL</td>
<td>37-48</td>
<td>05</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>DOWN TIME</td>
<td>49-60</td>
<td>05</td>
<td>12</td>
</tr>
<tr>
<td>MODERATOR</td>
<td>WEIGHT/QUANTITY IN CORE</td>
<td>1-24</td>
<td>06</td>
<td>24</td>
</tr>
<tr>
<td></td>
<td>AVG. TEMP</td>
<td>25-34</td>
<td>06</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>MAX. TEMP</td>
<td>35-44</td>
<td>06</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>45-54</td>
<td>06</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>55-64</td>
<td>06</td>
<td>10</td>
</tr>
<tr>
<td>BLANKET GAS</td>
<td></td>
<td>1-14</td>
<td>07</td>
<td>14</td>
</tr>
</tbody>
</table>

*Vol 14 No 2 June 1967*
Thus search for the shape of a reactor core would mean searching the first three elements of each column in the matrix for data on reactor core. Since the correspondence between an element or elements of the matrix and a specific head is known beforehand, search and listing are greatly facilitated. Further, it is possible to build up this correspondence between specific heads and the elements of the matrix as a machine dictionary. At the time of search it is possible to program the computer to ascertain first the major head under which the item of data occurs and then also the element(s) that constitute the data itself. Once these are ascertained specific searches can be made.

The data pertaining to each major head (for all reactors) was read into the memory as a matrix. The matrix representing data pertaining to each major head was given a mnemonic name, such as GENER for 'General Information', CORE for 'Core data' and so on.

Transfer to Disk.

Once the data was read into memory, transfer to disk was accomplished as follows:

A specific sector of the disk was made ready to be written upon. (A sector of the disk is a unit portion of the disk on which can be stored 100 digits or 50 alphameric characters). The writing of data began from this sector and was continued in sectors having addresses in an ascending sequence, until all the data was stored. With the knowledge of the address of the first sector and the number of elements of a matrix that were stored in any one sector, it was possible to pull out from the disk and put into memory any specific portion or portions of the data pertaining to any major head either for search or for listing.

The block diagram given in Fig. 1 represents broadly the method followed in storing data on disks.

Retrieval Programmes

The actual manipulation of a file for retrieval is preceded by an analysis of the question and decision as to the search strategy to be adopted. In a computer system, the strategy is generally translation into a programme. Once a generalized programme for a type of search strategy is made it can be used, with minor alterations, for other searches of a similar nature.

In this experiment a user's profile has been assumed. Attempt has been made to include search programmes based on a few search strategies.

The search strategies for which programmes were written are:
1. Single aspect search;
2. Logical product search;
3. Between the limits search; and
4. Search for upper limit of an item of data.

Single Aspect Search

The question chosen to write a programme for this kind of search was: What are the names of reactors whose core shape is a cylinder?

In order to retrieve names of reactors with core shape equal to a cylinder, it was necessary to define a variable (which was named 'Quest') and make it value equal to 'Cylinder'. The portion of the core data for each reactor which contained information on its shape was fetched from the disk. The shape of each was compared with the value of Quest. Whenever there was an equivalence, a branch was made and name of the reactor was printed out. This proceeded until the core shape of all reactors in the file were searched.

The block diagram for this type of search is given in Fig. 2. The results of this programme are presented in appendix I.

The programme written for the single aspect search could be used for similar searches with alterations in the definition of the variable 'Quest' and in comparison statements depending upon the number of elements in the matrix to be compared with the value of Quest.

The Logical Product Search

The question chosen for writing and testing a programme for this type of search was: What are the names of the Reactors, which are Pressurized Light Water Cooled and in operation in the U.S.A.?

In order to retrieve names of reactors satisfying both the criteria given above the search had to first proceed as follows:

Two variables had to be defined say Q 1 and Q 2. The variable Q 1 was given the value 'Pressurized Light Water' and Q 2, was given...
the value 'U. S. A.' These variables were then read into memory. The General Information pertaining to each reactor was fetched (since coolant and country are to be found under the head General Information) from the disk into memory. The elements constituting the coolant of a reactor was compared with the variable \( Q_1 \); on equivalence, the elements of the matrix constituting the country in which the reactor was located was compared with \( Q_2 \). Again on equivalence, a branch was made and the name of the reactor was printed out. Thus only when both conditions were satisfied the name was printed out.

The block diagram for this search is given in Fig. 3. A portion of the programme written in Fortran-II is given in appendix 2. The results of this search are presented in appendix 3.

The programme written for the logical product search could be used for similar searches with alterations in the definition of the variables \( Q_1 \) and \( Q_2 \), and in comparison statements depending upon the number of elements in the matrix to be compared with values of \( Q_1 \) and \( Q_2 \).

Between the Limits Search Programme

The question chosen for writing and testing a programme for this type of search was: What are the names of the reactors whose net efficiency falls in the range of 20 and 30 per cent?

That is \(20\% \leq \text{net efficiency} \leq 30\%\).

In order to retrieve names of reactors satisfying the above condition, the method adopted was as follows:

The two limits, viz: 20 and 30 were made equal to two variables \( \text{LIMI} \) and \( \text{LIMZ} \). These variables were read into the memory. The General Information pertaining to each reactor was fetched from the disk since net efficiency is a specific head under General Information. The net efficiency of each reactor was first compared with the lower limits. If it was found to be greater, or equal a branch was made and the net efficiency was compared with the upper limit. If it was lesser or equal, a branch was made and the names were printed out as search results. If either one of these was not satisfied, the search proceeded to the net efficiency of the next reactor and so on until the entire file was searched.

The block diagram for this type of search is given in Fig. 4. The results of the search are presented in the appendix 4.

Search Programme for Upper Limit of an Item of Data

The question chosen for writing and testing a programme for this type of search was:

What is the name of the reactor in your file with the maximum rated output. Also give details pertaining to its core and fuel element?

The method followed was as follows:

The General Information for all reactors was fetched into memory. The element(s) constituting the rated output of the first reactor was compared with that of the second reactor. The greater of the two was made equal to a variable which was named \( \text{EMAX} \). \( \text{EMAX} \) was compared with the rated output of the next reactor. The greater of the two replaced the former \( \text{EMAX} \). This continued until all the reactors in the file were searched. The value of the final \( \text{EMAX} \) gave the highest rated output of any reactor in the file. The name, core and fuel element information pertaining to this was then printed out.

The results are given in appendix 5.

Programmes for the Listing of Data

As a by product, a data retrieval system using computers could produce frequent compilations of data. It was with this view that programmes for the listing of data was attempted. Any attempt at listing of stored data has its implications at every stage of work. For instance, it is necessary at the time of encoding to record the data as it is found in the source. In other words codes for items of data cannot be adopted. If they are adopted, they will have to be decoded before they are listed or printed. This involves much programming effort, not commensurate with the frequency of need for such listing.

In this experiment the listing has been done only for a few major heads for some reactors.

The work involved in this aspect of the experiment may be divided as:

1. Decision as to the layout of the lists; and
2. Programming to produce lists in accordance with the decided layout.

Decision as to the layout becomes necessary if the computer print outs have to be duplicated and circulated. It was decided to have a two column layout for the listing of data.

It was decided to start the subheads at the first indentation and the specific heads at the second indentation. In this way the context in which the specific heads occurred would be given by the subhead.

The programming for listing of data involved the fetching from the disk of data under a major head and printing out of these according to the specified format. Great care had to be taken to see that the format conformed rigidly with the decisions taken earlier.

The print out from the computer have been given in the appendix 6.

Conclusions

The above experiment is at best of an exploratory nature. The finer aspects of file organisation and file search have to be investigated. The method of storage adopted in this experiment is wasteful of disk storage, though it offers certain advantages at the time of search. Attempts should be made to evolve generalised methods of file storage and retrieval for specific facts or data.

Acknowledgements

The authors are grateful to the Director, Insdoc for providing computer facilities to them for the above work.

Block Diagram for Storage of Data on Disk

![Block Diagram for Storage of Data on Disk](image-url)
COMPUTERISED DATA RETRIEVAL

Block Diagram for Single Aspect Search Programme
(Aspect 'QUEST' is being searched in the file)

Figure 2

Block Diagram for Logical Product Search
(File Search for Aspects QUEST 1 and QUEST 2)

Figure 3.
Block Diagram for between the Limits Search Programme
(QUEST is being searched for limits LIM 1 and LIM 2)

Figure 4.

APPENDIX

SINGLE ASPECT SEARCH PROGRAMME RESULTS

REACTORS WITH CORE SHAPE = CYLINDER

F+RST ATOMIC POWER STATION OF THE USSR
VORONEZH ATOMIC POWER STATION
DRESDEN NUCLEAR POWER STATION

1. Single Aspect Search Results
2. Programme for Logical Product Search
LOGICAL PRODUCT SEARCH PROGRAMME RESULTS

QUESTION. WHAT ARE THE NAMES OF REACTORS THAT ARE PRESSURIZED LIGHT WATER COOLED AND IN OPERATION IN USA

CONSOLIDATED EDISON THORIUM REACTOR

SHIPPINGPORT ATOMIC POWER STATION

STATIONARY MEDIUM POWER STATION

YANKEE ATOMIC ELECTRIC COMPANY

STOP

3. Logical Product Search Results

BETWEEN THE LIMITS SEARCH PROGRAMME RESULTS

REACTORS WITH NET EFFICIENCY BETWEEN TWENTY AND THIRTY PERCENT

BELGIAN THERMAL REACTOR BR 3

CONSOLIDATED EDISON THORIUM REACTOR

SHIPPINGPORT ATOMIC POWER STATION

VORONEZH ATOMIC POWER STATION

YANKEE ATOMIC ELECTRIC COMPANY

DRESDEN NUCLEAR POWER STATION

STOP

4. Between the Limits Search Results
COMPUTERISED DATA RETRIEVAL

SHIPPINGPORT ATOMIC POWER STATION

CORE DATA

SHAPE CYLINDRICAL

DIMENSIONS

DIAMETER/BREADTH 6.8 FT
HEIGHT 6 FT
LENGTH

NO OF CHANNELS

FUEL
CONTROL ROD
FUEL ELEMENT
SAFETY ROD

NO OF SUBASSEMBLIES 32
FUEL ROD
STATIONARY
MOVEABLE

LATTICE

SHAPE
PIITCH
FUEL ROD:
SUBASSEMBLIES
SPACING 6 INS

CRITICAL MASS 28.7 KG U235
AT TEMPERATURE 525 DEG F

FUEL ELEMENT

FORM SANDWICH PLATE TYPE
AT SEED BLANKET SEED

DIMENSIONS

DIAMETER
INNER
OUTER
LENGTH 71.75 IN
BREADTH 2.05 IN
HEIGHT/THICKNESS 0.039 IN

NO. OF PELLETS/ROD/TUBE

EXPERIENCE

DIAMETER
INNER
OUTER
LENGTH

NO. OF PLATES/RODS/TUBES-
PER ELEMENT 60

FORM PELLETS ROD TYPE
AT SEED BLANKET BLANKET

DIMENSIONS

DIAMETER
INNER
OUTER
LENGTH 0.358 IN
BREADTH 0.349 IN
HEIGHT/THICKNESS

NO. OF PELLETS/Rod/TUBE

EXPERIENCE

DIAMETER
INNER
OUTER
LENGTH

5. Upper Limit Search Results
SHIPPINGPORT—ATOMIC POWER STATION

GENERAL INFORMATION

REACTOR TYPE
ENRICHMENT (PERCENTAGE) 93
MODERATOR LIGHT WATER
COOLANT PRESSURIZED LIGHT WATER

RATED OUTPUT PER REACTOR
GROSS HEAT 225 MW
GROSS ELECT 67 MW
NET ELECT 60 MW
SELF CONSUMPTION 11.7

NO OF REACTORS 1
NET EFFICIENCY (PERCENTAGE) 26.6

LOCATION (COUNTRY) USA

CONSTRUCTION SCHEDULE
REACTOR CRITICAL DEC. 1957
FULL POWER OPERATION DEC. 1957

CORE DATA

SHAPE CYLINDRICAL

DIMENSIONS
DIAMETER/BREADTH 6.8 FT
HEIGHT 6 FT
LENGTH

NO OF CHANNELS
FUEL
CONTROL ROD
FUEL ELEMENT
SAFETY ROD

NO OF SUBASSEMBLIES 32
FUEL ROD
STATIONARY
MOVEABLE

LATTECE SHAPE

5. Listing of Data