



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

ADA138200

A COMPUTER COMPATIBLE SYSTEM FOR THE CATEGORIZATION,
 ENUMERATION, AND RETRIEVAL OF NINETEENTH AND EARLY TWENTIETH
 CENTURY ARCHAEOLOGICAL MATERIAL CULTURE

by Randolph E. Donahue

Accession for	
NTIS GPO	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/ _____	
Availability Codes	
Dist	Availability and/or Special
A/1	



Submitted to

National Park Service
 Mid-Atlantic Region
 143 South Third Street
 Philadelphia, PA 19106

Submitted by

The Museum
 Michigan State University
 East Lansing, MI 48824

DTIC
ELEC
S FEB 21 1984 D

December 1983

Chris Chelms

DISTRIBUTION STATEMENT A
 Approved for public release:
 Distribution Unlimited

33272-101

REPORT DOCUMENTATION PAGE	1. REPORT NO.	2.	3. Recipient's Accession No. ADA138200
4. Title and Subtitle A COMPUTER COMPATIBLE SYSTEM FOR THE CATEGORIZATION, ENUMERATION, AND RETRIEVAL OF NINETEENTH AND EARLY TWENTIETH CENTURY ARCHAEOLOGICAL MATERIAL CULTURE		5. Report Date December 1983	
7. Author(s) Charles Cleland, Randolph Donahue, W. Lee Minnerly, and		6.	
9. Performing Organization Name and Address The Museum, Michigan State University East Lansing, Michigan 48824		8. Performing Organization Rept. No.	
12. Sponsoring Organization Name and Address U.S. Department of the Interior for U.S. Army Corps of Engineers National Park Service 143 South Third Street Philadelphia PA 19106		10. Project/Task/Work Unit No.	
15. Supplementary Notes Mobile, Alabama 36628		11. Contract(G) or Grant(G) No. (C)CX4000-3-0005 (G)	
13. Type of Report & Period Covered Final		14.	
16. Abstract (Limit: 200 words) This report presents a system for classifying nineteenth century and early twentieth century material culture from archaeological contexts. It consists of three parts, a codebook, a codebook manual, and a computer software package. The codebook provides a system for classifying artifacts and artifact fragments according to material of manufacture and form, organized to segregate material, style, and manufacturing techniques of functional and chronological significance. The codebook manual contains instructions for making critical classification decisions as well as illustrations and references to facilitate the process. The software manual is not compatible with the current version of the codebook but provides an illustration of the utility of computer assisted enumeration and retrieval.			
17. Document Analysis a. Descriptors b. Identifiers/Open-Ended Terms c. COSATI Field/Group			
Availability Statement DISTRIBUTION STATEMENT A Approved for public release; Distribution Unlimited		19. Security Class (This Report) unclassified	21. No. of Pages Part 3:25
		20. Security Class (This Page)	22. Price

(See ANSI-Z39.18)

See Instructions on Reverse

OPTIONAL FORM 272 6-77
(Formerly NTIS-35)
Department of Commerce

Abstract

This report presents a system for classifying nineteenth century and early twentieth century material culture from archaeological contexts. It consists of three parts, a codebook, a codebook manual, and a computer software package. The codebook provides a system for classifying artifacts and artifact fragments according to material of manufacture as well as form, organized to segregate material, style, and manufacturing techniques of functional and chronological significance. The codebook manual contains instructions for making critical classification decisions along with illustrations and references to facilitate the process. The software manual is not compatible with the current version of the codebook but is an illustration of the utility of computer assisted enumeration and retrieval.

Preface and Acknowledgements

Recognizing that the excavation at the extinct nineteenth century townsites of Colbert, Barton and Vinton, Mississippi would result in a huge return of artifacts, as well as complex data manipulation problems representative from the Tombigbee Historic Townsites Project, the Park Service-Mid Atlantic Region and the Corps of Engineers-Mobile determined to design and produce a system which would facilitate the classification and computer manipulation of recovered artifacts.

It was initially hoped that this system would be available in time to use for the classification of material from other nineteenth century sites being excavated within the Tombigbee River Multi-Resource District of the Tennessee-Tombigbee Waterway. Unfortunately this was not to be since the design and testing of the system took far longer than was originally anticipated. While it was recognized from the beginning that the system was to represent the first step in a classification and retrieval scheme which would evolve through use, the difficulties of producing a working system were underestimated.

Essentially, the system as it now stands has evolved through three phases. The first was the design of the codebook and software. The second was a test of the system by coding the artifacts recovered from Colbert, Barton, and Vinton. The third, in part concurrent with the second phase, was the redesign of the codebook, preparation of the manual, and fine tuning of the retrieval software. In fact, the lessons learned in coding pointed out some major problems with the categorization and as a result, the codebook was ultimately refined three times. Since the Colbert, Barton and Vinton material was coded under the first and second versions, artifacts coded from these sites are not in conformity with the present system. Further, the software package presented here has not been updated to work with the present version of the codebook and manual although it is compatible with the second version of the codebook published in the project's Phase II report (Minnerly 1983). It is included here to illustrate the capability of computer assisted retrieval. It is hoped that appropriate changes will soon be made in the software package to make it compatible with the present codebook as well as to make it more powerful and efficient in operation.

Designing a universal system for classifying all of nineteenth century material culture that can be used by cataloguers with minimal training is not an easy task. This attempt has had its vocal critics; among these are a number of experts in nineteenth century material culture who have devoted a great deal of energy in reviewing drafts of the codebook and manual. For their criticisms and many useful suggestions, great thanks is due to Bill Adams, Marley Brown, Olive Jones, George Miller, Randy Moir and Tef Roddeffer. It should be understood that these colleagues do not necessarily agree with the categorization suggested, the contents of the manual, nor even the feasibility of the system itself.

Of the Tombigbee Historic Townsites staff, special thanks is due to Bob Sonderman and Randy Donahue who devoted many hours beyond their formal association with the project and to Tia Maxwell, project artist who is responsible for the illustrations in the manual. W. Lee Minnerly, who spearheaded the project to begin with, was instrumental in developing most of the major concepts as well as contributing a major portion of the section on ceramics in the manual. Lee's important contributions are gratefully acknowledged.

Finally and most especially, thanks is due to Kim McBride, editor, typist, coordinator and all around dynamo who is responsible for getting the job done.

C. Cleland
East Lansing
December 10, 1983

Foreword

This volume is the third and final part of the categorization, enumeration, and retrieval system, and is meant to be used in conjunction with Part I, the codebook, and Part II, the codebook manual. However, recent revisions to the codebook and manual, but not the software, present incompatibilities. The version of the codebook to which this software is compatible is published along with the Phase II report of the Tombigbee Historic Townsites Project (Oral Historical, Documentary, and Archaeological Investigations of Barton and Vinton, Mississippi: An Interim Report on Phase II of the Tombigbee Historic Townsites Project, edited by W. Lee Minnerly). Despite the incompatibilities occasioned by recent changes in the codebook and manual, the manual itself should still be of some use with the earlier version of the codebook.

A further warning to the reader concerns the structure of the data files, which renders the program expensive to run, and the need for revisions to facilitate analytical research. Suggestions for improvement appear in Section 7.0 of this report.

TABLE OF CONTENTS

Abstract.	1
Preface and Acknowledgements	11
Foreword.	iv
List of Tables	vi
1.0 INTRODUCTION	1
2.0 USE OF THE AIS	1
3.0 SYSTEM CONTROL STATEMENTS.	1
3.1 INPUT AND OUTPUT VIA TERMINAL	1
3.2 PROCEDURE FOR BATCH OUTPUT VIA TERMINAL	2
3.3 BATCH INPUT AND OUTPUT	3
4.0 PROGRAM COMMAND STATEMENTS	3
5.0 RETRIEVAL OF DATA FILES FROM TAPE	4
5.1 RETRIEVAL OF FILES FROM TAPE VIA BATCH	4
5.2 RETRIEVAL OF FILES FROM TAPE BY TERMINAL.	6
6.0 TECHNICAL INFORMATION AND PROGRAMMERS	6
6.1 PROGRAM EXC	6
6.2 PROGRAM CRFL	9
6.3 PROGRAM SITES	9
6.4 PROGRAM HEADERS	9
6.5 LOCATION OF PROGRAM AND DATA FILES AT MSU	10
7.0 SUGGESTED FUTURE IMPROVEMENTS FOR THE AIS	10
APPENDICES	
Appendix A, Section 1, Example Runs	11
Appendix A, Section 2, Program Listings.	16
Appendix B, Contents of Site Files Structures.	see microfiche
Appendix C, Definition of Material Codes	25
Appendix D, Artifact Codes and Acronyms.	see microfiche

LIST OF TABLES

1. Sequence of program commands	5
2. Program files	7
3. Data files	8

1.0 INTRODUCTION

The Tombigbee Historic Townsites Project Archaeological Information System (AIS) was developed between 1980 and 1982 by Dr. Robert I. Wittick and Mr. Randolph E. Donahue of Michigan State University. The AIS is an interactive retrieval system for site and artifact information collection by the Tombigbee Historic Townsites Project between 1979 and 1981, and it was developed for use on the Michigan State University Control Data Corporation Cyber 170 series, model 750, computer. Although the system is highly portable to other computer facilities, all instructions in this report refer to its use at MSU.

The AIS manipulates hierarchically structured data files and prints out the desired information according to the set of commands presented by the user. The AIS program prompts the user for each required piece of information, thus making it a very easy retrieval system to use even by those unfamiliar with computers. Given the size of the files being manipulated, the AIS is very rapid and can process data as fast as the terminal can print the information. Options for having the data stored for later printing or for having the data printed at a high speed "batch" printer are also presented in this report.

The data are structured in a hierarchical arrangement of ten levels: spatial provenience data include site, unit, stratigraphic level, feature, subfeature, and area information, and artifact typology data include material category, material subcategory, artifact category information. Section 4.0 presents various selection options for these ten levels.

2.0 USE OF THE AIS

The AIS was written with emphasis on convenient interactive use. Nevertheless, it can be used from batch (card input) or from a terminal with the output disposed to batch for high speed printing. The procedure for using the AIS can be divided into two parts, system control statements that tell the computer what to do, and program command statements that tell the AIS program what information the user wants.

3.0 SYSTEM CONTROL STATEMENTS

Users of the CDC computer at Michigan State are required to have a valid problem number and for batch use, a problem number card (PNC). Applications are available from the Computer Laboratory, Room 220, Computer Center.

3.1 INPUT AND OUTPUT VIA TERMINAL

This is the most convenient but most expensive way to use the AIS system. If a large quantity of information is being retrieved,

one of the alternative approaches is suggested. The user must first link up the terminal with the computer such as by telephone and modem; the current telephone number is 353-8555. Direct line terminals are also available in Room 208, Computer Center. The user will be requested to enter the password, problem number, and user identification; this information is presented to the user when the problem number application is returned. The user should remember that at a terminal, every line entered must be followed by the RETURN key. Once successfully logged in, the computer will prompt the user with READY and the time. The following lines should be entered.

```
RTL,60.
ATTACH,AIS,AIS,PW=AIS.
AIS.
```

The user now begins interacting with the AIS program, where program commands will be requested. When the AIS is stopped and the user is completely finished, the following statement should be entered:

```
LOGOUT,T.
```

3.2 PROCEDURE FOR BATCH OUTPUT VIA TERMINAL

At times the user will only have access to a CRT (nonprinting) terminal or will have a large amount of output to be printed. Using the program for batch output will resolve the first problem and reduce time and cost of the second. The following control statements must be entered after logging-in:

```
RTL,60.
PROMPT.
ATTACH,AIS,AIS,PW=AIS.
AIS,DATA.
```

Program command statements are entered here as in Section 3.1. Only asterisks will be printed to inform the user that input is required.

```
REWIND,DATA.
DISPOSE,DATA,PR=B.
LOGOUT,T.
```

A sequence number is printed after the Dispose statement. This number is used to locate the output, which is filed according to the last three digits in Room 208, Computer Center. Be sure to copy this number down.

With this method, the user has various options for handling the output. Various alternative dispositions are available as well as cataloging the output for later printing. These alternative means

for handling the output file are discussed in the Michigan State University User's Guide to the Interactive System.

3.3 BATCH INPUT AND OUTPUT

If the user is not in a hurry for the output, this approach can be most cost effective. The user submits the system control statements and program command statements with computer cards. The deck for performing this routine should contain the following cards:

```
Sequence Card
PNC (Problem Number Card).
Job Card
Password Card
ATTACH,AIS,AIS,PW=AIS.
AIS.
7/8/9
.
.
.
Command statements
.
.
.
6/7/8/9
```

The sequence card is provided in Room 208, Computer Center, and contains the number the output is filed under in that room. The next card is the problem number card and the following card contains the user's identification starting in column 1, followed by a set of parameters, separated by commas, that set limits to the potential cost of the run. An average run will require the following parameters:

```
Id ,CM60000,JC500,L50,T30,RG2.
```

Discussion of these parameters is found in the Michigan State University Guide to the Scope/Hustler System. The password card contains PW= "User's Password" starting in column 1. The 7/8/9, 6/7/8/9 cards contain these numbers multipunched in column 1.

4.0 PROGRAM COMMAND STATEMENTS

The same set of questions is always presented to the user unless either material categories I or II is selected. The user should have a good understanding of allowable commands at each level of the hierarchy and should know exactly which commands are required for any given run. This alone will eliminate almost all possible errors. After the program begins executing, it will request information for each hierarchic level. Table 1 presents a brief description of allowable inputs at each level. It should be remembered that only for input and output at a terminal (Section 3.1) will actual questions

be presented requesting the next piece of information. Thus, one must know the order and number of commands required by the other methods. The command sequence follows that found in Table 1.

After requesting the type of header record, the information will be printed at the terminal unless batch output was requested. In all cases, however, the system returns to the beginning and requests a site designation or "STOP." If the user is finished, then by entering STOP the user will be exited from program control to computer system control. Alternatively, the user can continue retrieving information by entering a site designation. Examples of various ways of requesting information are presented in Appendix A.

5.0 RETRIEVAL OF DATA FILES FROM TAPE

The Tombigbee Historic Townsites Project data files contain a large quantity of site and artifact information. This massive amount of data is stored at MSU in two ways: on disk and on magnetic tape. Disk storage is much more convenient than tape storage because it allows immediate access to the data when one requests the file on which that information is found. But because disk storage is very expensive for large quantities of data, tape storage is a valuable option. However, the user must know how to transfer information from tape to disk.

The user will discover whether the information desired is on disk after entering the program command of the site number. If not available, the computer will respond with "REQUESTED SITE (#) NOT ON SYSTEM." Be sure that the site number wanted is the one requested and not a typographical error. To save costs, one should also be aware of which other sites will contain desired information so that all site data files can be requested at one time. Files once retrieved will be available for 15 days. There are two ways of retrieving files from tapes, by batch and by terminal.

5.1 RETRIEVAL OF FILES FROM TAPE VIA BATCH

The following cards are required. A description of most of them is presented in Section 3.2.

```
Sequence Card
PNC
Id ,NT1,JC2000,CM60000,RG2,T10.
PW= Password
PFLOAD,NT=1933, I=INPUT
7/8/9
.
.
.
```

A list of all site numbers desired prefixed by the letter E and beginning in Column 1. Example:

Table 1. Sequence of program commands.

Hierarchic level of input request	Range of acceptable values	Location of input values
Site	4 digit integer of STOP	Appendix B
Unit	1 to 3 digit integer or ALL	Appendix B
Level	1 or 2 digit integer or ALL	Appendix B
Feature	1 to 3 digit integer or ALL	Appendix B
Subfeature	1 to 3 digit integer or ALL	Appendix B
Area	1 digit integer or ALL	Appendix B
Material Category	Roman numeral between I and XII or ALL	Appendix C
Material Subcategory (requested only if material category is I or II)	letter A to E or ALL	Appendix D
Artifact Category	2 digit integer or ALL	Appendix D
Artifact Subcategory	3 digit integer or ALL	Appendix D
Full Header Records	YES or NO	

E4999
 .
 .
 .
 6/7/8/9

Be sure to include the parameter NT=1 in the Jobcard or the program will not retrieve the files.

5.2 RETRIEVAL OF FILES FROM TAPE BY TERMINAL

To perform this task at the terminal, the user should first log in and then enter the following control statements:

```
SYSTEM BATCH.
N,100.
*JOBCARD*,T10,RG2,NT1,JC2500.
PFLOAD,NT=UP1993,I=INPUT
*EOS
.
.
.
```

Each site desired is entered line by line prefixed by the letter E. Example:

```
E5200
.
.
.
*EOI
=
GO.
LOGOUT,T.
```

A sequence number will be presented after "GO." It refers to a printout filed in Room 208, Computer Center. It will inform the user if the retrieval was successful.

6.0 TECHNICAL INFORMATION AND PROGRAMMERS

AIS version 1.0 and all peripheral programs are written in CDC Fortran version 5.0, an extended version of Fortran 77 described in ANSI document X.9-1978. A brief description of each program important to the operation and maintenance of the AIS is presented below. Tables 2 and 3 provide some brief but critical information concerning the program and data files.

6.1 PROGRAM EXC

This is the heart of the retrieval system, and it is the only program required by users of the AIS (see Appendix A, Section 2).

Table 2. Program files.

Program Name	Permanent File Name	Passwords	Status
EXC	AIS	CN=TOM EX=BIG MD=BEE TK-AIS	Binary file
EXC	AISEW	TK-TOM	Editor work file
CRFL	ACRONYMPROGRAM	TK-TOM	Coded file
SITES	LGO SITES	TK-TOM	Relative file
SITES	SITSEW	TK-TOM	Editor work file
SITES	SITES	TK-TOM	Coded file
HEADERS	LGO HEADERS	TK-TOM	Relative file
HEADERS	HEADERLISTPROGRAM	TK-TOM	Coded file

Table 3. Data files.

Permanent File Name	Passwords	Status	Program(s) Input	Program Output	Contents
TAPE-FILE PN-MUI5916	--	--	DUMPN	--	System derived file for manipulating tapes.
ACRONYMS	TK-TOM	Coded, Sequential access file	CRFL	--	Artifact codes and acronyms.
TAPE2ACRONYMSFILE	TK-TOM	Coded,	EXC	CRFL	Artifact codes and acronyms.
TAPE3ACRONYMSFILE	TK-TOM	Coded, Random access file	EXC	CRFL	Index, artifact code, and material code.
E____ (four digit site file #)	None	Coded, Sequential access file HEADERS	EXC	--	Artifact codes and header records.

Three data files are manipulated by the program: TAPE1, the excavation data; and TAPE2 and TAPE3, which contain information allowing for the writing of the acronyms. These two data files are random accessed files produced by Program CRFL (Section 6.2). Program EXC requires approximately 50K and can process approximately 250 data records per second of central processing time. This varies greatly depending on the frequency of header records.

The main program attaches necessary files, reads the data file, determines what kind of line of code is presented, which subroutine it should use for interpreting that line, and prints the output file. Subroutine PRIHD is used for printing complete headers while Subroutine PRIPHD is used for printing partial headers.

Subroutine PARMS requests command statements. It uses Subroutine FIND to insure the value entered will be equivalent to that requested by counting the number of characters entered. It then uses Subroutine PACK to combine the characters into a single character string variable.

Subroutine HEADER decodes the header statements with the use of Subroutine PACK, which recombines characters into a character string variable. Subroutine CODER deciphers the artifact categories and subcategories. It also uses Subroutine PACK to recombine the characters into a character string variable. Subroutine MAJOR interprets material category codes. Subroutine LETTER interprets the material subcategory codes if Roman numeral I or II is selected at the level of material category (See Appendix A, Section 2).

6.2 PROGRAM CRFL

This program produces the random access tape files used by Program EXC from a sequential file containing the artifact codes and their corresponding acronyms. TAPE1 is the acronyms input file while TAPE2 and TAPE3 are the random access output files (see Table 3). The program flow is self-explanatory. See Appendix A, Section 2 for a listing of the program.

6.3 PROGRAM SITES

This program combines data of selected unit numbers from input files TAPE3 into a single output file TAPE4. The unit numbers are requested from input at the start of the program. This program allows users to combine data from numerous files into a single file, which can then be catalogued. The flow of this program is self-explanatory. A listing can be found in Appendix A, Section 2.

6.4 PROGRAM HEADERS

This very simple program reads through a data file (TAPE3) and writes out all header records on TAPE4 in the same format. The site number (file number) is printed at the top of each listing. An indefinite number of files can be processed at one time. The flow

of this program is self-explanatory. A copy of the program is found in Appendix A, Section 2.

6.5 LOCATION OF PROGRAM AND DATA FILES AT MSU

All program files and data files are located on nine track tapes UP1992 and UP1993. Files were originally manipulated between tapes and disk using MSU system routine DUMPN. For information concerning DUMPN and the corresponding retrieval program GETPF, refer to the following on-line documentation:

```
HELP,L*UNSUP,F*DUMPN.  
HELP,L*UNSUP,F*GETPF.
```

PFLOAD commands are adequate for retrieving these files as well.

7.0 SUGGESTED FUTURE IMPROVEMENTS FOR THE AIS

Many of the following suggestions result from discussions the author has had with Dr. Robert I. Wittick, Director of User Services, MSU Computer Laboratory. Dr. Wittick wrote most of the Program EXC and produced an efficient routine for retrieving coded data and presenting that information in a form understandable to anybody wanting to work with it. On the other hand, the data structure was designed elsewhere and is difficult to manipulate and decode. This has kept the overall system far below its potential level of cost-effectiveness.

The one major change suggested here is to restructure the data into a fixed format structure, perhaps with code numbers in field 1 to indicate the type of data record presented. This would require modification of Program EXC (and most others as well). It is estimated that the size of Program EXC could be reduced 50% or more and its efficiency at least doubled. The change in the data structure would also allow for a 50% reduction in storage costs because the current structure has each artifact category code and its frequency on separate records; these can easily be shifted to one record.

Another important addition to the AIS is a data entry program that would significantly reduce errors, thus reducing editing costs. This program would be a very simple program to write. There is one other very necessary program that would require much work. This is a program that would restructure requested data into a format accepted by major analytical packages like SPSS, Clustan and Geosys. Without such a program, researchers are limited to obtaining data listings, exactly what a retrieval system does, but not adequate for scientific research. While discussed as separate programs, there is no reason why these additional routines cannot be overlays or subroutines of Program EXC. It should be remembered that every program written to manipulate the data files will be costly to run as long as the files remain in their current structure.

APPENDIX A, Section 1

Example Run

TOTAL COUNT FOR UNIT: 54.

SITE: 4942 PROVENIENCE: Y UNIT: 167 N CORD: 788 E CORD: 354
 LEVEL: 2 FEATURE: 0 SUBFEATURE: 0 AREA: 0
 SCREEN METHOD: 12 EXCAVATORS INITIALS: CH DATE: 07/01/80

MATERIAL: I (GLASS)
 ARTIFACTS COUNT
 17. 12 (3.) PLATE WINDOW GLASS FRAG
 49. 33 (1.) CLR NPB UNID BOT/JAR BODY FRAG UNMARKD
 49. 113 (3.) GRN UNID BOT/JAR BODY FRAG UNMARKD
 62. 122 (1.) PB UNDEC UNID TABLWARE FRAG BODY
 62. 123 (1.) PB UNDEC UNID TABLWARE FRAG BASE/STEM

MATERIAL: II (CERAMIC)
 ARTIFACTS COUNT
 2. 22 (2.) GLAZED SOFT FIRED BRICK FRAG
 2. 32 (15.) UNGLAZED SOFT FIRED BRICK FRAG
 14. 2 (1.) EDGE DEC WPE BODY
 29. 12 (1.) WHITE CAST UNDEC PLAIN WPE BODY
 43. 22 (1.) UNDEC SLIP CLAY GLAZE CPS BODY
 43. 22 (1.) UNDEC SLIP CLAY GLAZE CPS BODY
 46. 1 (1.) UNDEC AND GLAZED PORCELAIN RIM

MATERIAL: III (METAL)
 ARTIFACTS COUNT
 11. 202 (2.) UNID MACHINE CUT NAIL FRAG
 21. 92 (2.) IRON/STEEL FRAG

TOTAL COUNT FOR UNIT: 35.

SITE: 4942 PROVENIENCE: Y UNIT: 169 N CORD: 390 E CORD: 320
 LEVEL: 1 FEATURE: 0 SUBFEATURE: 0 AREA: 0
 SCREEN METHOD: 1 EXCAVATORS INITIALS: JU DATE: 07/01/80

MATERIAL: I (GLASS)
 ARTIFACTS COUNT
 17. 12 (4.) PLATE WINDOW GLASS FRAG
 49. 83 (1.) 008 UNID BOT/JAR BODY FRAG UNMARKD

MATERIAL: II (CERAMIC)
 ARTIFACTS COUNT
 2. 32 (1.) UNGLAZED SOFT FIRED BRICK FRAG
 29. 12 (1.) WHITE CAST UNDEC PLAIN WPE BODY

TOTAL COUNT FOR UNIT: 7.

SITE: 4942 PROVENIENCE: Y UNIT: 170 N CORD: 389 E CORD: 348
 LEVEL: 1 FEATURE: 0 SUBFEATURE: 0 AREA: 0
 SCREEN METHOD: 1 EXCAVATORS INITIALS: MB JL DATE: 07/01/80

MATERIAL: I (GLASS)
 ARTIFACTS COUNT
 3. 11 (1.) GLASS BUTTON SEM-THROUGH COMP
 9. 22 (2.) GLASS LAMP CHIMNEY FRAG
 17. 12 (7.) PLATE WINDOW GLASS FRAG
 39. 24 (1.) CLR PS BOT/JAR LPNK FRAG UNID
 39. 74 (1.) GRN NPB UNID BOT/JAR LPNK FRAG UNID
 49. 23 (7.) CLR PS UNID BOT/JAR BODY FRAG UNMARKD
 49. 33 (17.) CLR NPB UNID BOT/JAR BODY FRAG UNMARKD
 49. 63 (2.) GRN PS UNID BOT/JAR BODY FRAG UNMARKD

MSU-AIS EXCAVATION SYSTEM
VERSION 1.0

SPECIFY SITE NO. (OR *STOP*):

SPECIFY SEARCH PARAMETERS:

UNIT?
LEVEL?
FEATURE?
SUBFEATURE?
AREA?
MATERIAL CATEGORY?
ARTIFACT CATEGORY?
ARTIFACT SUBCATEGORY?
PRINT COMPLETE HEADERS?

SITE: 4942 PROVENIENCE: Y UNIT: 37 N CORD: 388 E CORD: 382
LEVEL: 1 FEATURE: 0 SUBFEATURE: C AREA: 0
SCREEN METHOD: 1 EXCAVATORS INITIALS: SM DATE: 02/06/80

MATERIAL: I (GLASS)
ARTIFACTS COUNT
49. 73 (1.) GRN NPB UNID BOT/JAR BODY FRAG UNMARKD

MATERIAL: II (CERAMIC)
ARTIFACTS COUNT
2. 32 (47.) UNGLAZED SOFT FIRED BRICK FRAG

MATERIAL: XI (MINERAL)
ARTIFACTS COUNT
3. 8 (8.) CHARCOAL/COAL

TOTAL COUNT FOR UNIT: 49.

SITE: 4942 PROVENIENCE: Y UNIT: 167 N CORD: 388 E CORD: 364
LEVEL: 1 FEATURE: 0 SUBFEATURE: 0 AREA: 0
SCREEN METHOD: 1 EXCAVATORS INITIALS: DM DATE: 07/01/80

MATERIAL: I (GLASS)
ARTIFACTS COUNT
17. 12 (2.) PLATE WINDOW GLASS FRAG
49. 93 (3.) ASA UNID BOT/JAR BODY FRAG UNMARKD
49. 113 (4.) GRN UNID BOT/JAR BODY FRAG UNMARKD

MATERIAL: II (CERAMIC)
ARTIFACTS COUNT
2. 32 (22.) UNGLAZED SOFT FIRED BRICK FRAG
29. 11 (2.) WHITE CAST UNDEC PLAIN WPE RIM
29. 12 (2.) WHITE CAST UNDEC PLAIN WPE BODY
29. 13 (1.) WHITE CAST UNDEC PLAIN WPE BASE
32. 2 (1.) UNID MWPE BODY
37. 12 (2.) UNDEC GLAZED OT CPE BODY
43. 12 (1.) UNDEC SALT VAPOR GLAZE CPS BODY
43. 22 (1.) UNDEC SLIP CLAY GLAZE CPS BODY

MATERIAL: III (METAL)
ARTIFACTS COUNT
9. 171 (1.) KEYHOLE ESCUTCHEON COMP
11. 81 (2.) 60 MACHINE CUT NAIL
11. 121 (1.) 1 0 MACHINE CUT NAIL
11. 202 (3.) UNID MACHINE CUT NAIL FRAG
13. 162 (1.) OTHER MISC HARDWARE FRAG
21. 42 (9.) IRON/STEEL FRAG

1. 12 (1.) MODIFIED LITHIC FRAG
 1. 22 (1.) DEBRIS FRAG
 1. 42 (1.) FIRE CRACKED ROCK FRAG

TOTAL COUNT FOR UNIT: 367.

SITE: 4942 PROVENIENCE: Y UNIT: 170 N CORD: 304 E CORD: 349
 LEVEL: 2 FEATURE: 0 SUBFEATURE: 0 AREA: 0
 SCREEN METHOD: 1 EXCAVATORS INITIALS: WB JL DATE: 07/01/80

MATERIAL: I (GLASS)
 ARTIFACTS COUNT
 3. 11 (3.) GLASS BUTTON SEW-THROUGH COMP
 17. 12 (2.) PLATE WINDOW GLASS FRAG
 39. 34 (1.) CLR NPB BOT/JAR LPMK FRAG UNID
 49. 23 (6.) CLR PS UNID BOT/JAR BODY FRAG UNMARKD
 49. 33 (7.) CLR NPB UNID BOT/JAR BODY FRAG UNMARKD
 59. 35 (2.) CLR NPB UNID BOT/JAR BASE FRAG, UNID
 61. 22 (1.) PB PRESSED-MOLDED OT VSSL FRAG BODY

MATERIAL: II (CERAMIC)
 ARTIFACTS COUNT
 2. 22 (3.) GLAZED SOFT FIRED BRICK FRAG
 2. 32 (61.) UNGLAZED SOFT FIRED BRICK FRAG
 16. 32 (1.) PAINTED POLYCHROME WPE BODY
 17. 42 (1.) PRINTED PURPLE WPE BODY
 29. 12 (4.) WHITE CAST UNDEC PLAIN WPE BODY
 37. 12 (2.) UNDEC GLAZED OT CPE BODY
 40. 2 (1.) UNDEC WPS BODY
 43. 22 (1.) UNDEC SLIP CLAY GLAZE CPS BODY
 46. 1 (1.) UNDEC AND GLAZED PORCELAIN RIM

MATERIAL: III (METAL)
 ARTIFACTS COUNT
 11. 91 (1.) 70 MACHINE CUT NAIL
 11. 101 (1.) 80 MACHINE CUT NAIL
 11. 131 (1.) 120 MACHINE CUT NAIL
 11. 202 (73.) UNID MACHINE CUT NAIL FRAG
 13. 161 (1.) OTHER MISC HARDWARE COMP
 13. 162 (1.) OTHER MISC HARDWARE FRAG
 16. 172 (1.) TABLESPOON FRAG

MATERIAL: IV (BONE)
 ARTIFACTS COUNT
 12. 2 (1.) UNMODIFIED FRAG

MATERIAL: VII (STONE)
 ARTIFACTS COUNT
 6. 2 (6.) SLATE BOARD FRAG

MATERIAL: XII (PREHISTORIC)
 ARTIFACTS COUNT
 1. 42 (1.) FIRE CRACKED ROCK FRAG

TOTAL COUNT FOR UNIT: 184.

SITE: 4942 PROVENIENCE: Y UNIT: 171 N CORD: 372 E CORD: 366
 LEVEL: 1 FEATURE: 0 SUBFEATURE: 0 AREA: 0
 SCREEN METHOD: 1 EXCAVATORS INITIALS: LS DATE: 07/01/80

MATERIAL: I (GLASS)

49. 73	(2.)	GRN NPB UNIC BOT/JAR BODY FRAG UNMARKD
49. 83	(1.)	OGG UNID BOT/JAR BODY FRAG UNMARKD
49. 83	(3.)	OGG UNID BOT/JAR BODY FRAG UNMARKD
49. 113	(3.)	GRN UNID BOT/JAR BODY FRAG UNMARKD
59. 35	(2.)	CLR NPB UNID BOT/JAR BODY FRAG UNMARKD
59. 95	(1.)	CLR NPB UNID BOT/JAR BASE FRAG, UNID
61. 131	(1.)	AGA UNID BOT/JAR BASE FRAG, UNID
62. 123	(2.)	NPB UNDEC ST VSSL FRAG RIM/NECK
64. 22	(4.)	PR UNDEC UNID TABLEWARE FRAG BASE/STEP
			CLR PS UNID GLS FRAG UNDEC

MATERIAL: II (CERAMIC)

ARTIFACTS	COUNT	
2. 22	(4.)
2. 32	(178.)
8. 192	(1.)
16. 23	(1.)
17. 92	(7.)
29. 11	(1.)
29. 12	(21.)
29. 13	(2.)
40. 1	(2.)
40. 2	(1.)
40. 4	(1.)
43. 12	(1.)
43. 22	(6.)
43. 23	(1.)
46. 1	(1.)

GLAZED SOFT FIRED BRICK FRAG
UNGLAZED SOFT FIRED BRICK FRAG
UNDEC/UNMARKD STONEWARE DETACH STEM PIPE FRAG
PAINTED OT MONOCHROME WPE BASE
PRINTED FLOWED WPE BODY
WHITE CAST UNDEC PLAIN WPE RIM
WHITE CAST UNDEC PLAIN WPE BODY
WHITE CAST UNDEC PLAIN WPE BASE
UNDEC WPS RIM
UNDEC WPS BODY
UNDEC WPS OT
UNDEC SALT VAPOR GLAZE CPS BODY
UNDEC SLIP CLAY GLAZE CPS BODY
UNDEC SLIP CLAY GLAZE CPS BASE
UNDEC AND GLAZED PORCELAIN RIM

MATERIAL: III (METAL)

ARTIFACTS	COUNT	
7. 222	(2.)
7. 222	(1.)
8. 302	(1.)
11. 41	(1.)
11. 51	(9.)
11. 61	(4.)
11. 71	(2.)
11. 81	(11.)
11. 101	(9.)
11. 111	(2.)
11. 121	(2.)
11. 202	(221.)
13. 22	(1.)
16. 192	(1.)
21. 42	(14.)

OTHER CONTAINER FRAG
OTHER CONTAINER FRAG
WIRE STAPLE FRAG
20 MACHINE CUT NAIL
30 MACHINE CUT NAIL
40 MACHINE CUT NAIL
50 MACHINE CUT NAIL
60 MACHINE CUT NAIL
80 MACHINE CUT NAIL
90 MACHINE CUT NAIL
1 D MACHINE CUT NAIL
UNID MACHINE CUT NAIL FRAG
CHAIN FRAG
IRON/STEEL FRAG

MATERIAL: IV (BONE)

ARTIFACTS	COUNT	
12. 2	(3.)

UNMODIFIED FRAG

MATERIAL: V (SHELL)

ARTIFACTS	COUNT	
3. 2	(6.)

UNMODIFIED FRAG

MATERIAL: X (PLASTIC)

ARTIFACTS	COUNT	
7. 1	(1.)

OTHER COMP

MATERIAL: XII (PREHISTORIC)

ARTIFACTS	COUNT	
1. 11	(1.)

MODIFIED LITHIC COMP

APPENDIX A, Section 2

Program Listings

```

PROGRAM EXC(INPUT,OUTPUT,T:PE1,T:PE2,T:PE3)
CHARACTER*1 I(1:8),FROM,DEM
CHARACTER*4 CODE,EX,N,KCODE,KLETT
CHARACTER*3 DATE
CHARACTER*6 SITE
CHARACTER*10 TITLE(1:2)
CHARACTER*40 ACRO
COMMON/C/ CTYPE
COMMON/HEAD/ISITE,IUNIT,INC,IEC,LEVEL,IFEA,ISUE,IAPEA,ISM
COMMON/HEAD/PROV,EX,DATE
COMMON/REC/KLEV,KFEC,KSUB,KAREA,KM,KN,KUNIT
COMMON/REC/KCODE,KLETT
DATA TITLE/'GLASS','FRANIC','METAL','PHONE','SMELL','MCCO/VEGARL',
* 'STONE','LEATHER','RJOBRE','PLASTIC','MINERAL','PREHISTORIC'/
PRINT *
4 FORMAT('CMSU-AIS EXCAVATION SYSTEM'/Y,'VERSION 1.0')
OPEN(2,ACCESS='DIRECT',RECL=7)
CLOSE(2,STATUS='DELETE')
OPEN(3,ACCESS='DIRECT',RECL=3)
CLOSE(3,STATUS='DELETE')
ERRSR('ATTACH TO FILE: I:ACRO','FILL','MSTON')
OPEN(2,ACCESS='DIRECT',RECL=7)
OPEN(3,ACCESS='DIRECT',RECL=3)
PRINT *
10 FORMAT('SPECIFY SITE NO. (OR *STOP*):')
READ(3,SITE(1:3))
FORMAT(A4)
3 IF(SITE(2:2).EQ.*S*)STOP
TOT=0
IFLG=0
SITE(1:1)=*E*
OPEN(1)
CLOSE(1,STATUS='DELETE')
ERRSR('ATTACH TO FILE: I:ACRO','FILL','MSTON')
IF(ERF.T..160 TO 2)
OPEN(1)
RCKIAD
CALL SARMS(1)
PRINT *,*PRINT COMPLETE HEADERS?*
READ(7,F7M)
7 FORMAT(A1)
READ(1,1,END=1000)A
FORMAT(1X,6(A1))
CALL HEAD(1)
HEAD=1
IF(IFLG.EQ..160 TO 1)
IF(SUM.GT..1)PRINT *,SUM
16 FORMAT('TOTAL COUNT FOR UNIT: ',F7.0)
SUM=0
IFLG=1
IF(KUNIT.EQ.-1.00.*UNIT.EQ.IUNIT)GO TO 45
GO TO 40
45 IF(KLEV.EQ.-1.00.*KLEV.FO.LEVEL)GO TO 20
HEAD=1
GO TO 10
50 IF(KFEA.EQ.-1.00.*KFEA.EQ.IFEA)GO TO 51
GO TO 40
51 IF(KSUM.EQ.-1.00.*KSUM.EQ.ISUM)GO TO 52
GO TO 40
52 IF(KAPEA.EQ.-1.00.*KAREA.EQ.IAPEA)GO TO 53
GO TO 40
53 IFNO=0
15 READ(1,1,END=1000)A
CALL MAJOR(A,LL)
IF(L.EQ.-1)GO TO 5
IF(L.EQ.-1)GO TO 60
IF(CODE(1:1).EQ.*A*.GR.KCODE.EQ.CODE)GO TO 55
CALL J1
GO TO 60
IF(J.EQ.-1)GO TO 20
READ(1,1)A
CALL LETTER(A(1),B)
IF(B.EQ.-1)GO TO 30
IF(I.EQ.-1.C) I=AJ.EQ..160 TO 20
IF(KLETT.EQ..1111.00.*KLETT.EQ..5)GO TO 62
IF(J.EQ.-1)
GO TO 20

```



```

4     CALL PACK(A,J,I,LEVEL)
      J=I-1
      GO TO 1000
7     CALL PACK(A,J,I,IFF=)
      J=I-1
      GO TO 1000
6     CALL PACK(A,J,I,ISU=)
      J=I-1
      GO TO 1000
9     CALL PACK(A,J,I,IAPE=)
      J=I-1
      GO TO 1000
10    CALL PACK(A,J,I,IS4)
      J=I-1
      GO TO 1000
11    KK=I-J
      IF (KK.NE.2.AND.KK.NE.4) STOP 2
      DO 11 L=1,KK
101   X(IN(L))=A(M)
      J=I-2
      GO TO 1000
12    KK=20-L-1,KK
      DO 12 L=1,KK
120   DAT(L)=A(M)
      GO TO 1100
1-CC
1100  RETURN
      END
SUBROUTINE PACK(A,J,I,INTX)
CHARACTER*1 A(50),ONE
CHARACTER*2 TWO
CHARACTER*3 THREE
CHARACTER*4 FOUR
CONTINUE
1     K=I-1
      IF (K.GT.4) STOP 1
      IF (K.EQ.1) GO TO 50
      GO TO (10,20,30,40),K
10    ONE=A(J)
      READ(ONE,11)INTX
      FORMAT(11)
20    TWO=L-1
      DO 20 L=1,2
21    TWO(L)=A(M)
      READ(TWO,22)INTX
      FORMAT(12)
      RETURN
30    THREE=L-1
      DO 30 L=1,3
31    THREE(L)=A(M)
      READ(THREE,33)INTX
      FORMAT(13)
      RETURN
40    FOUR=L-1
      DO 40 L=1,4
41    FOUR(L)=A(M)
      READ(FOUR,42)INTX
      FORMAT(14)
      RETURN
50    INTX=0
      RETURN
      END
SUBROUTINE CDEB(L,M,N)
CHARACTER*1 A(50)
IF (A(1).EQ.' ') GO TO 999
J=1
K=1
DO 20 I=1,50
IF (A(I).EQ.' ') GO TO 5
IF (A(I).NE.' ') GO TO 20
CALL PACK(A,J,I,M)
J=I-1
K=1
DO 20 I=1,50
IF (A(I).EQ.' ') GO TO 999
CALL PACK(A,J,I,M)
RETURN

```

1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30
 31
 32
 33
 34
 35
 36
 37
 38
 39
 40
 41
 42
 43
 44
 45
 46
 47
 48
 49
 50
 51
 52
 53
 54
 55
 56
 57
 58
 59
 60
 61
 62
 63
 64
 65
 66
 67
 68
 69
 70
 71
 72
 73
 74
 75
 76
 77
 78
 79
 80
 81
 82
 83
 84
 85
 86
 87
 88
 89
 90
 91
 92
 93
 94
 95
 96
 97
 98
 99
 100
 101
 102
 103
 104
 105
 106
 107
 108
 109
 110
 111
 112
 113
 114
 115
 116
 117
 118
 119
 120
 121
 122
 123
 124
 125
 126
 127
 128
 129
 130
 131
 132
 133
 134
 135
 136
 137
 138
 139
 140
 141
 142
 143
 144
 145
 146
 147
 148
 149
 150
 151
 152
 153
 154
 155
 156
 157
 158
 159
 160
 161
 162
 163
 164
 165
 166
 167
 168
 169
 170
 171
 172
 173
 174
 175
 176
 177
 178
 179
 180
 181
 182
 183
 184
 185
 186
 187
 188
 189
 190
 191
 192
 193
 194
 195
 196
 197
 198
 199
 200

```

15 CALL PACK(A,J,I,N)
RETURN
2: CONTINUE
RETURN
9: N=N-1
RETURN
END
SUBROUTINE MAJOR(A,LL)
COMMON /C/CCOE
CHARACTER*4 ROMAN(12),CCOE
CHARACTER*1 A(40)
DATA ROMAN/'I','II','III','IV','V','VI','VII','VIII','IX',
* 'X','XI','XII'/
K=1
CODE=0
DO 2:0 I=1,4:
IF(A(I),N=0) GO TO 1:9
IF(N.EC.0) GO TO 2:
N=N-I
IF(N.GT.0) GO TO 3:75
DO 4: N=L-1,N
N=L-J-1
15 CODE(L:L)=A(N)
GO TO 3:00
1:9 IF(N.EC.0) J=I
K=1
CONTINUE
2:0 DO 3:0 I=1,12
3:0 IF(CCOE.EQ.ROMAN(I)) GO TO 4:00
CONTINUE
4:0 RETURN
L=L-1
RETURN
END
SUBROUTINE LETTER(A,R)
CHARACTER*1 LET(4),A,R
DATA LET/'A','S','C','D','E','F'/
DO 1:0 I=1,6
IF(I.EQ.LET(I)) GO TO 5:0
2: CONTINUE
N=N+1
RETURN
5:0 RETURN
END
SUBROUTINE FIND(A,J,I)
CHARACTER*1 A(60)
J=1
I=0
DO 1:0 K=1,6:0
IF(A(K).E.C.0) AND(L.EQ.0) GO TO 9
IF(A(K).E.C.0) GO TO 5
1:0 RETURN
L=1
GO TO 1:0
5: J=J+1
CONTINUE
END
SUBROUTINE FARMS(A)
CHARACTER*1 A(60)
CHARACTER*2 KCODE,KLETT
COMMON /REQ/KLEVV,KPEA,KSUB,KAREA,KM,KK,KUNIT
COMMON /RECCH/RCCOE,KLETT
KLEVV=1
KPEA=1
KSUB=1
KAREA=1
KUNIT=1
KMS=1
KNS=1
PRINT *,'SPECIFY SEARCH PARAMETER:'
PRINT *,' UNIT:'
READ 33,A
IF(A(1).E.C.'A') GO TO 5:0
CALL FIND(A,J,I)
CALL PACK(A,J,I,KUNIT)
PRINT *,' LEVEL:'
FORMAT(1:1)
READ 33,A

```

15 CALL PACK(A,J,I,N)
 RETURN
 2: CONTINUE
 RETURN
 9: N=N-1
 RETURN
 END
 SUBROUTINE MAJOR(A,LL)
 COMMON /C/CCOE
 CHARACTER*4 ROMAN(12),CCOE
 CHARACTER*1 A(40)
 DATA ROMAN/'I','II','III','IV','V','VI','VII','VIII','IX',
 * 'X','XI','XII'/
 K=1
 CODE=0
 DO 2:0 I=1,4:
 IF(A(I),N=0) GO TO 1:9
 IF(N.EC.0) GO TO 2:
 N=N-I
 IF(N.GT.0) GO TO 3:75
 DO 4: N=L-1,N
 N=L-J-1
 15 CODE(L:L)=A(N)
 GO TO 3:00
 1:9 IF(N.EC.0) J=I
 K=1
 CONTINUE
 2:0 DO 3:0 I=1,12
 3:0 IF(CCOE.EQ.ROMAN(I)) GO TO 4:00
 CONTINUE
 4:0 RETURN
 L=L-1
 RETURN
 END
 SUBROUTINE LETTER(A,R)
 CHARACTER*1 LET(4),A,R
 DATA LET/'A','S','C','D','E','F'/
 DO 1:0 I=1,6
 IF(I.EQ.LET(I)) GO TO 5:0
 2: CONTINUE
 N=N+1
 RETURN
 5:0 RETURN
 END
 SUBROUTINE FIND(A,J,I)
 CHARACTER*1 A(60)
 J=1
 I=0
 DO 1:0 K=1,6:0
 IF(A(K).E.C.0) AND(L.EQ.0) GO TO 9
 IF(A(K).E.C.0) GO TO 5
 1:0 RETURN
 L=1
 GO TO 1:0
 5: J=J+1
 CONTINUE
 END
 SUBROUTINE FARMS(A)
 CHARACTER*1 A(60)
 CHARACTER*2 KCODE,KLETT
 COMMON /REQ/KLEVV,KPEA,KSUB,KAREA,KM,KK,KUNIT
 COMMON /RECCH/RCCOE,KLETT
 KLEVV=1
 KPEA=1
 KSUB=1
 KAREA=1
 KUNIT=1
 KMS=1
 KNS=1
 PRINT *,'SPECIFY SEARCH PARAMETER:'
 PRINT *,' UNIT:'
 READ 33,A
 IF(A(1).E.C.'A') GO TO 5:0
 CALL FIND(A,J,I)
 CALL PACK(A,J,I,KUNIT)
 PRINT *,' LEVEL:'
 FORMAT(1:1)
 READ 33,A

```

IF(A(1).EQ.'A')GO TO 110
CALL FIND(1,J,I)
CALL PACK(A,J,I,KLEV)
100 PRINT 'SURFEATURE?'
READ 33,A
IF(A(1).EQ.'A')GO TO 110
CALL FIND(1,J,I)
CALL PACK(A,J,I,KFEA)
110 PRINT 'SURFEATURE?'
READ 33,A
IF(A(1).EQ.'A')GO TO 120
CALL FIND(1,J,I)
CALL PACK(A,J,I,KSUH)
120 PRINT 'AREA?'
READ 33,A
IF(A(1).EQ.'A')GO TO 130
CALL FIND(1,J,I)
CALL PACK(A,J,I,KRFA)
130 PRINT 'MATERIAL CATEGORY?'
READ 3,KCODE
FOR(I=1,4)
KLETT=ALL
IF(KCODE.NE.'')AND(KCODE.NE.'1')GO TO 140
PRINT 'MATERIAL SUBCATEGORY?'
140 READ 3,KLETT
PRINT 'ARTIFACT CATEGORY?'
IF(A(1).EQ.'A')GO TO 150
CALL FIND(1,J,I)
CALL PACK(A,J,I,KM)
150 PRINT 'ARTIFACT SUBCATEGORY?'
READ 33,A
IF(A(1).EQ.'A')GO TO 200
CALL FIND(2,J,I)
CALL PACK(A,J,I,KM)
200 CONTINUE
RETURN
END
SUBROUTINE PRIND
COMMON /HEAD/SITE,UNIT,INC,IEC,LEVEL,IFEA,ISUB,IAREA,ISM
COMMON /HEAD/PROV,EXIN,DATE
CHARACTER*1 PROV
CHARACTER*4 EXIN
CHARACTER*4 DATE
PRINT 1, SITE, PROV, UNIT, INC, IEC
FOR(I=1,3) I= IEC, IE, IEC
PROVENIENCE: ',A1,' UNIT: ',I4,
* ' E CORD: ',I4, ' E CORD: ',I4)
PRINT 2, LEVEL, IFEA, ISUB, IAREA
FOR(I=1,4) I= IFEA, ISUB, IAREA, IFEA
SCREEN: ',I4, ' FEATURE: ',I4, ' SURFEATURE: ',
* I4, ' AREA: ',I4)
PRINT 3, ISIN(1:2), EXIN(1:4), DATE(1:2), DATE(3:4), DATE(5:6)
FOR(I=1,4) I= EXIN(1:4), EXIN(5:8), EXIN(9:12), EXIN(13:16)
SCREEN: ',I4, ' EXCAVATOR INITIALS: ',
* I4, ' DATE: ',I4, '/',I4, '/',I4)
RETURN
END
SUBROUTINE PRIND
COMMON /HEAD/SITE,UNIT,INC,IEC,LEVEL,IFEA,ISUB,IAREA,ISM
PRINT 1, UNIT, LEVEL, IFEA, ISUB, IAREA
FOR(I=1,4) I= IFEA, ISUB, IAREA, IFEA
SCREEN: ',I4, ' LEVEL: ',I4, ' FEATURE: ',I4,
* SURFEATURE: ',I4, ' AREA: ',I4)
RETURN
END

```

14:07:43 08/09/82 SL*1396 387 Lines print.
7 Pages print. Cost at 2.97 is 2.22


```

PROGRAM SITES(INPUT,OUTPUT,TAPE1=INPUT,TAPE2=OUTPUT,TAPES,TAPES)
CHARACTER*3 UNIT(100)
CHARACTER*5 SITE
CHARACTER*6 A(10)
WRITE(2,220)
220 FORMAT('ENTER THE NUMBER OF CELLS TO BE ENTERED IN IJ FORMAT:')
AC(1)=100/NUNIT
240 FORMAT('I')
DO 10 I=1,NUNIT
WRITE(2,230)
230 FORMAT('CELL')
READ(1,100)UNIT(I)
100 FORMAT('I')
CONTINUE
WRITE(2,250)
250 FORMAT('SPECIFY SITE NUMBER OR *STOP*')
READ(1,110)SITE(2:5)
110 FORMAT('I')
IF(SITE(2:5).EQ.*S*)STOP
OPEN(3)
CLOSE(3,STATUS='DELETE')
SITE(1:1)=*S*
WRITE(2,120)SITE
120 FORMAT('ATTACH',3,SITE)
IF(IPR.GT.0)GO TO 8
CPE(3)
300 READ(3,300)C(3)A
FORMAT('X,10A6')
IF(A(1)(5:5).EQ.**)GO TO 15
IF(IPR.NE.1)GO TO 6
WRITE(4,300)A
GO TO 6
15 I=1
NCHAR=0
IF(A(2)(4:4).EQ.**)NCHAR=NCHAR+1
IF(A(2)(3:3).EQ.**)NCHAR=NCHAR+2
DO 20 I=1,NUNIT
IF(A(2)(2:NCHAR).EQ.UNIT(I))GO TO 7
CONTINUE
IM=0
GO TO 6
END

```

14:56:52 02/09/82 SLE1396 42 Lines print.
3 Pages print. Cost at \$63 to 2.06

```

PROGRAM HEADERS (INPUT,OUTPUT,TAPE1=INPUT,TAPE2=OUTPUT,TAPE3,TAPE*)
CHARACTER*5 SITE
CHARACTER*10 ALPHA(6)
WRITE(2,200)
200 FORMAT(100,'COPY SITE NUMBER OR "STCP"')
READ(1,100) SITE(2:5)
100 FOR(1:10)
IF(SITE(2:2).EQ.'C')STCP
OPEN(1)
CLOSE(1,STATUS='DELETE')
SITE(1:1)='C'
PRINT(1,ATTACH=3,SITE)
IF(ENR.GT.0)GO TO 5
OPEN(3)
WRITE(6,400)SITE(2:5)
400 FORMAT(11,'A',//)
READ(3,300,END='ALPHA
300 FORMAT(11,'E',//)
IF(ELPHA(1)EQ.'A')HE...GO TO 2
WRITE(6,41)ALPHA
FOR(1:6)ALPHA
GO TO 2
END

```

14:06:00 05/09/52 CL81396 24 Lines print.
3 Pages print. Cost at Pgs 18 1 0.00

APPENDIX C
DEFINITION OF MATERIAL CODES

<u>Material Codes</u>	<u>Material</u>
I	Glass
II	Ceramic
III	Metal
IV	Bone
V	Shell
VI	Wood/Vegetable
VII	Stone
VIII	Leather
IX	Rubber
X	Plastic
XI	Mineral
XII	Prehistoric

ATE
LMED
8