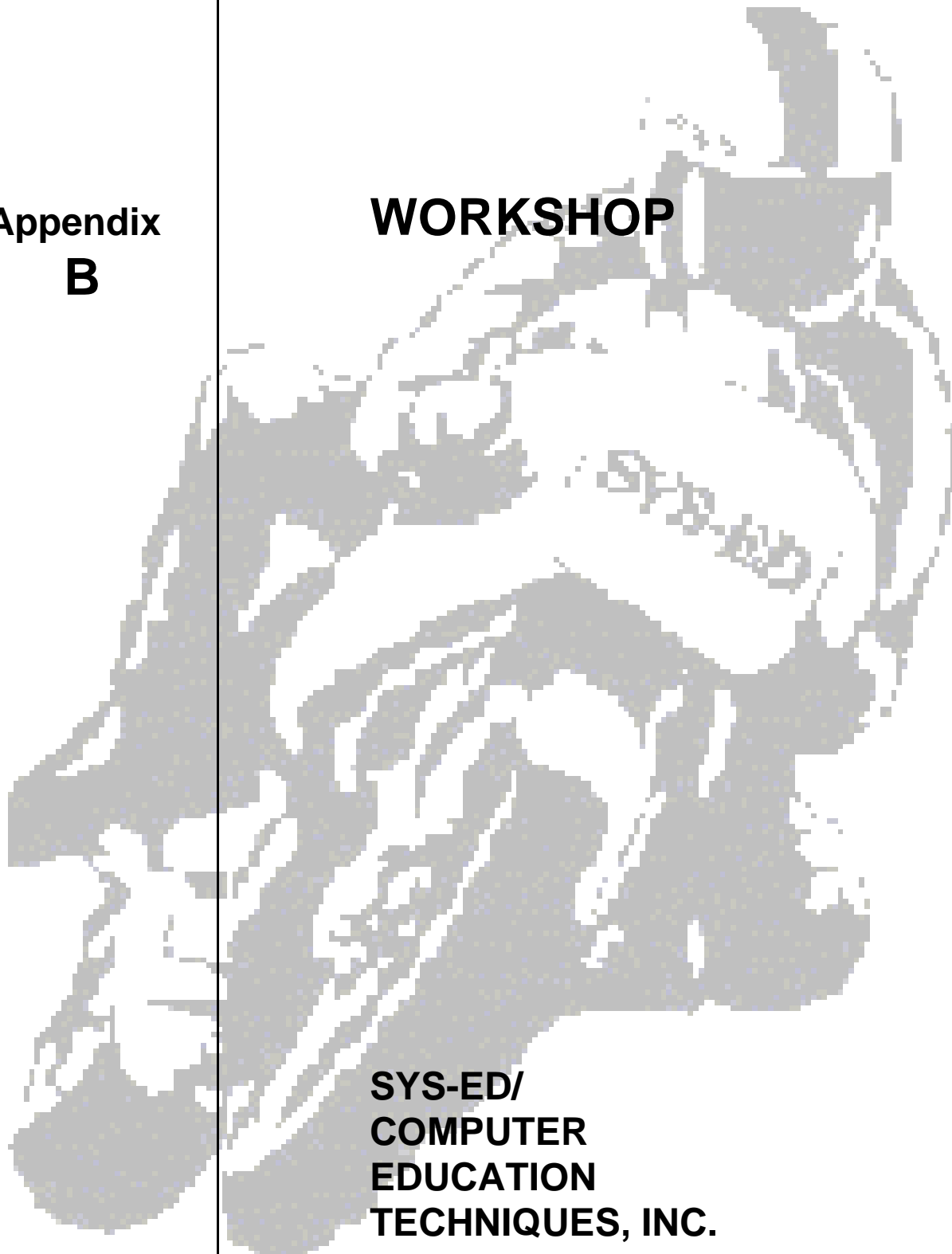


**Appendix
B**

WORKSHOP



**SYS-ED/
COMPUTER
EDUCATION
TECHNIQUES, INC.**

1 IDCAMS Processing**Objectives**

1. Print and create clusters that will be used as input to the future workshops.
2. List all data sets with a SYSED high level qualifier (or qualifier provided by the instructor).
3. Code and test IDCAMS JCL.
4. Code and test IDCAMS commands.

Instructions

1. Using the LISTCAT command, list all datasets names with a specific qualifier. Instructor will provide the qualifier.
2. Using the PRINT command:

Print in character format a member in a PDS.
Print in HEX format a sequential file.
Print the first 10 records in a dataset.
3. Create the following 3 ESDS data sets using the following attributes:

INPUT: qual.ITEMFIL.INPUT

Item number

3 bytes numeric **

Quantity

4 bytes numeric

Filler

73 bytes

80

INPUT: qual.SYSED.CUSTFIL.INPUT

Customer number

3 bytes numeric

Name

50 bytes alpha

State

2 bytes alpha

Filler

25 bytes

80

Input file:SYSED.CUST.FILE.MAINT
 Delete record:
 Type code 1 byte alpha 'D'
 Customer number 3 bytes numeric
 Filler 76 bytes
 80

Add record:
 Type code 1 byte alpha 'A'
 Customer number 3 bytes numeric
 Name 50 bytes alpha
 State 2 bytes alpha
 Filler 24 bytes
 80

Order record:
 Type code 1 byte alpha 'O'
 Customer number 3 bytes numeric
 Item number 3 bytes numeric
 Quantity ordered 4 bytes numeric
 Filler 69 bytes
 80

4. Using the REPRO command, copy data into each of the above ESDS datasets.
5. Print three new ESDS data sets.
6. List all data sets starting with qual using the LEVEL parameter.

Potential Problems

1. No datasets located in the printing the clusters.
2. Volume is not available.

2 RRDS Processing

Objectives

1. Define an RRDS cluster.
2. Program to load data into this cluster.

Instructions

1. Define RRDS cluster.

Name: qual.TEAM#.ITEM.FILE (replace # with your team number)

Record size: 11 bytes

Maximum records: 1000

2. Code a program to read the input file and store in the new RRDS dataset. Edit the Item Number to insure that it contains a valid numeric.

INPUT: qual.ITEMFIL.INPUT

Item number 3 bytes numeric **

Quantity 4 bytes numeric

Filler 73 bytes

80

** This field also will be the Relative Record Number.

OUTPUT: qual.TEAM#.ITEM.FILE

Item number 3 bytes numeric

Quantity 4 bytes numeric

Quantity back ordered 4 bytes numeric (Zero fill)

11

Potential Problems

1. Must move Item number to Relative Key field.
2. Must define Relative Key field outside of record definition (ie. Define in Working Storage not File Section).
3. Designate Access as Dynamic or Random on output file.
4. Must have AS-ddname on input file in Select statement.

3 KSDS Processing - Load

Objectives

1. Define a KSDS cluster.
2. Program to load data into this cluster.

Instructions

1. Define cluster:

Name: qual.TEAM#.CUSTOMER.FILE
Record size: 18 to 59 characters
Maximum Number of Records: 3500
Free space: 10% CI and CA

2. Program to load only the header records.

INPUT: qual.SYSED.CUSTFIL.INPUT

Customer number	3 bytes numeric
Name	50 bytes alpha
State	2 bytes alpha
Filler	<u>25</u> bytes
	80

OUTPUT: qual.TEAM#.CUSTOMER.FILE

Header record	
Customer number	3 bytes numeric**
Suffix	2 bytes numeric**
State	2 bytes alpha
Name	50 bytes alpha
Number of orders	<u>2</u> bytes numeric (zero fill)
	59

Trailer record (will not be used until workshop 4)

Customer number	3 bytes numeric
Suffix	2 bytes numeric
State	2 bytes alpha
Item number	3 bytes numeric
Quantity	4 bytes numeric
Qty backordered	<u>4</u> bytes numeric

These two fields make up the key.

Use the 3 byte customer key coming in and append 2 zeros.

Potential Problems

1. Record Key must be defined as 5 bytes in record definition input area (File Section).
2. Must zero fill suffix field.
3. Must have AS-ddname on input file Select statement.

4 KSDS Processing - Update

Objectives

1. Update Customer data set.
2. Update Item data set.

Instructions

1. I-O files qual.TEAM#.ITEM.FILE
qual.TEAM#.CUSTOMER.FILE

Refer to Workshops 2 and 3 for ITEM record and HEADER record layouts.

Input file:SYSED.CUST.FILE.MAINT

Delete record:

Type code	1 byte alpha 'D'
Customer number	3 bytes numeric
Filler	<u>76</u> bytes
	80

Add record:

Type code	1 byte alpha 'A'
Customer number	3 bytes numeric
Name	50 bytes alpha
State	2 bytes alpha
Filler	<u>24</u> bytes
	80

Order record:

Type code	1 byte alpha 'O'
Customer number	3 bytes numeric
Item number	3 bytes numeric
Quantity ordered	4 bytes numeric
Filler	<u>69</u> bytes
	80

3. Processing
Keep total counts of:
orders
adds
deletes
duplicate adds
invalid deletes

Delete Processing:

On a Delete transaction append two zeros to customer number and delete the appropriate header record.

Keep a count of the number of deletes processed and the number of invalid deletes (customer not on file).

Hint: - The deletes are for headers only. There are no orders associated with headers to be deleted.

- There are invalid deletes.

Add Processing:

On an Add transaction append two zeros to customer number build and WRITE the appropriate header record.

Hint: - The adds are for header records only.

- There are duplicate adds.

Order Processing:

On an order transaction append two zeros to customer number. READ the header record and update the number of orders field. This number will become the suffix for the order record. REWRITE the header record. READ and update the Item file (subtract quantity ordered from quantity on the item record) Rewrite Item record. Build and WRITE trailer record.

Hint: - There are no orders for customers without headers.

- There are multiple orders for customers.
- For quicker coding ignore backorder fields.

Potential Problems

1. Not updating the number of orders field in the header record.

5 AIX Processing

Objectives

1. Define an Alternate Index.
2. Define a Path.
3. Build an Alternate Index.
4. Program to display the number of customers per state.

Instructions

1. Define AIX:
Name: qual.TEAM#.CUSTOMER.AIX
Record size:12 to 132
2. Define Path:
Name: qual.TEAM#.CUSTOMER.PATH
3. Build Alternate Index.
4. Code and test program to read customer file through the Path.

Potential Problems

1. Must define in the Select statement both a Record Key and an Alternate Record Key.

2. JCL must include two DD statements:

The first one must have DDname as in the Select statement.

Example: Select ... assign to MASTER from COBOL program

```
//MASTER DD DSN=TEAM#.CUSTOMER.FILE
```

Second one must have DDname appended with a 1

```
//MASTER1 DD DSN=TEAM#.CUSTOMER.PATH
```

Sample Solution Output:

CUST			CUST		
STATE	COUNT	ORDERS	STATE	COUNT	ORDERS
AL	1		MN	1	
AZ	1		MO	2	
CA	5	3	ND	1	
CO	4	10	NJ	2	
CT	1		NY	12	3
DC	1		OH	1	
DE	1		OR	1	
FL	1		PA	1	
GA	1		RI	1	5
HI	3	3	SD	1	9
IA	1		TN	1	
LA	1		TX	3	2
MA	2		WY	1	
MI	3				