

# MAFD4202 – Winter 2019

## Mainframe Development 1

### Team Programming Project

The Mainframe Development 1 team programming project is a group project that will create 4 interlinked COBOL programs. Groups are comprised of either 2-3 members that belong to the same class section. You will choose your group, BUT you will have to advise your professor by email of the names of your group members to confirm the group enrollment.

**No individual submissions will be accepted.** If you have not identified your group by 5pm on Friday, March 22nd, then you will be assigned to a group randomly. You will only be able to access the necessary input data once you have enrolled in a group.

This assignment is worth **20% of your overall grade.**

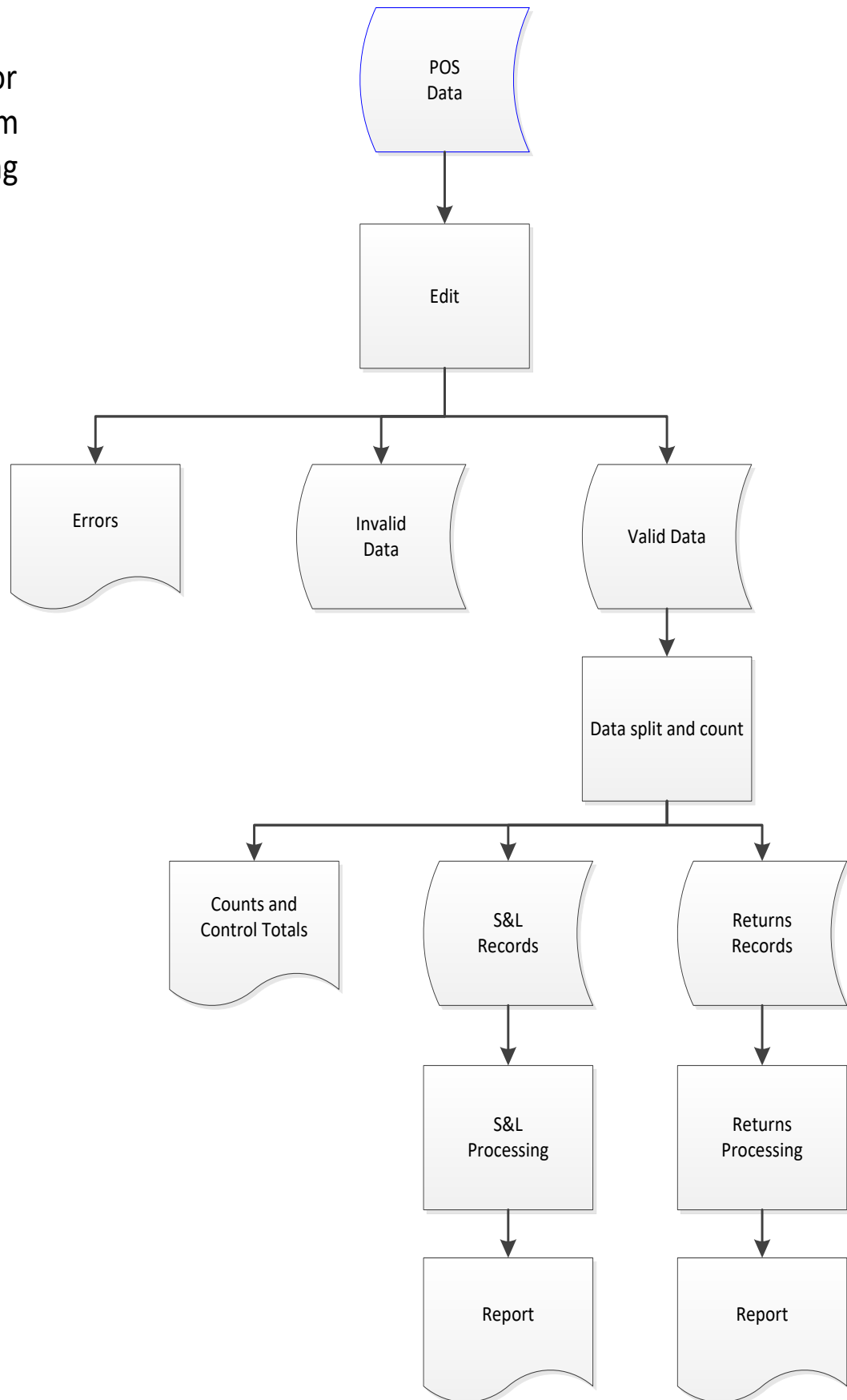
This system is to process sales records from the Point-of-Sale (POS) devices at our stores. There have 3 types of transactions (Sales, Returns and Layaways), and the store accepts Cash, Credit and Debit for payment types.

#### Contents

Flow of Data Through Project Deliverables .....	2
PROGRAM #1 – EDITS .....	3
Edit Rules.....	3
PROGRAM #2 – DATA SPLIT AND COUNT .....	5
TRANSACTION CODE 'S' or 'L' .....	5
TRANSACTION CODE 'R' .....	5
TRANSACTION CODE 'S' or 'L' or 'R' .....	5
PROGRAM #3 – S&L Processing .....	6
PROGRAM #4 – Type 'R' Processing.....	7
COMPLETION and SUBMISSION.....	8
EVALUATION – 100%.....	8
DUE DATE: Thursday, April 18 <sup>th</sup> , 8PM. ....	8

## Flow of Data Through Project Deliverables

Systems  
Flowchart for  
Cobol 1 Team  
Programming  
Project



## PROGRAM #1 – EDITS

The EDITS program is responsible for editing the input records to ensure validity. The record layout is:

Transaction Code	– position 1	– pic X.
Transaction Amount	– position 2-8	– pic 9(5)V99 (does not include taxes)
Payment Type	– position 9-10	– pic XX.
Store Number	– position 11-12	– pic XX or 99
Invoice Number	– position 13-21	– pic X(9)
SKU Code	– position 22-36	– pic X(15)

### Edit Rules:

1. All records must have a transaction code of either 'S', 'R' or 'L'.
2. The transaction amount field must be numeric
3. The Payment Type field must be 'CA', 'CR' or 'DB'
4. The Store Number must be one of '01', '02', '03', '04', '05', or '12'
5. The invoice number must be in the format XX-000000 where XX means the entry must be alphabetic and the 0 means numeric only.
6. The invoice number XX can only be A or B or C or D or E
7. The invoice number XX cannot have two letters the same
8. The invoice number 000000 cannot be greater than 900000 or less than 100000
9. All records should have a dash '-' in position 3.
10. The SKU code is a 15-character alphanumeric code that cannot be empty (spaces).

Loop through each record in the file and edit according to the rules above.

If a record is VALID (no edit errors) then the entire record is written onto the VALID RECORDS DATA FILE for later use.

See Systems flowchart.

If a record fails one or more edits then:

1. Write the entire record onto the INVALID ERRORS DATA FILE
2. Create an ERRORS report showing the record in error and as many errors as exist.

*No report layout is provided, design your own.*

You may wish to think about the way you wrote out the messages in Lab #7 for Bonus Amount, and/or follow a similar design as in Lab #8.

There are 10 business edit rules, so there could be a maximum of 10 errors per record.

3. For control purposes write out totals of how many records are read in, and the number of good and error records.

For example, if 30 records were input they might be 24 good records and 6 in error. Make sure these add up correctly.

## PROGRAM #2 – DATA SPLIT AND COUNT

Keep in mind that this program is only processing VALID data, so no error checking is required.

Since Program #1 copied whole data records the input data file SHOULD look the same as in Program #1. If you altered the record layout on the output of the first program then you will have to adjust accordingly.

The purpose of this program is basically to produce a report with some records counts and to split the records onto 2 data files. See the systems flowchart for how this relates to the other programs.

This program has no actual “Detail” output, all that is produced on the ‘Counts and Control Totals” Report is the totals listed above. Design your own report layout.

### TRANSACTION CODE ‘S’ or ‘L’

Transaction code ‘S’ is for a Sale, ‘L’ is for Layaway (which is considered a variation of a sale). Place these records on the S&L Records data file.

Produce the following totals:

Total number of S&L records (combined) and S&L Total Amount

Total number of ‘S’ records and Total Amount

Total number of ‘L’ records and Total Amount

Total transaction amount for each of the 6 stores listed above

% number of transactions in each Payment Type category

(for S&L only, type ‘R’ does not count)

### TRANSACTION CODE ‘R’

Type ‘R’ (returns) records go onto their own data file.

Produce the following totals:

Total number of ‘R’ records and Total Amount for each of the 6 stores

Total number of ‘R’ records and Total Amount

### TRANSACTION CODE ‘S’ or ‘L’ or ‘R’

Produce the following total:

Grand Total Amount, which is S&L Total Amount – ‘R’ total amount.

## PROGRAM #3 – S&L Processing

Since Programs #1 & #2 copied whole data records,  
the input data file SHOULD look the same as in Program #1 / #2.

The purpose of this program is basically to produce a detail report of sales.  
See the systems flowchart for how this relates to the other programs.

For each record on the input file produce on line of output showing all 6 input fields. Single space the output, 20 lines per page with page counter, report heading and column headers.

In the detail output, show the tax owing on that transaction (taxes are 13%).

**At the end of the Detail output, produce the following totals:**

- Total number of S&L records (combined) and S&L Total Amount

- Total number of 'S' records and Total Amount

- Total number of 'L' records and Total Amount

- The number and percentages of each of the Payment Types

  - 'CA' for cash, 'CR' for Credit Card or 'DB' for Debit

  - the 3 percentages should total 100%

- Total Tax Owing

- The Store number that has the highest S&L total transaction amount

- The Store number that has the lowest S&L total transaction amount  
(the last two are tricky until Arrays arrive in lecture)

Keep in mind that some of the Total numbers and amounts produced here must  
balance with the output from Program #2.

## PROGRAM #4 – Type ‘R’ Processing

Since Programs #1 & #2 copied whole data records,  
the input data file SHOULD look the same as in Program #1 / #2.

The purpose of this program is basically to produce a detail report of Returns.  
See the systems flowchart for how this relates to the other programs.

For each record on the input file produce on line of output showing all 6 input fields. Single space the output, 20 lines per page with page counter, report heading and column headers. Also, in the detail output show the tax that is now owed to us (taxes are 13%).

At the end of the Detail output write out the same ‘R’ total as Program #2 above:

Total number of ‘R’ records and Total Amount for each of the 6 stores.

Total number of ‘R’ records and Total Amount

Total Tax Owed to Us

Keep in mind that some of the Total numbers and amounts produced here must balance with the output from Program #2.

## COMPLETION and SUBMISSION

Only the input file is provided, so it is wise to get Program #1 running first to produce the other needed data files. Alternatively, you may want to take some time to manually generate the valid data file from Program #1, so that you can start working on other programs.

When you have the four programs running you can package the 4 programs as a single Visual Studio solution, or as four separate solutions. You may want to consider using a version control system for this, so that group members can gain easy access to your latest work.

You are to PRINT OUT AND HAND IN the 4 Reports shown in the Systems flowchart with a cover page showing group member's names and date.

Submit VS2017 solution(s) containing all 4 programs, data files, and reports. It is mandatory that a comment is included indicating who the actual author(s) of each of the programs are.

Each student will also submit a peer evaluation of themselves and the other members of their group to the DC Connect Dropbox. More explanation of this follows and will be discussed in class. Failure to submit a peer evaluation will result in a mark of 0 for that portion of the project.

## EVALUATION – 100%

Program #1	– 20%
Program #2	– 20%
Program #3	– 15%
Program #4	– 15%
Visual Studio Packaging and Configuration	– 15%
Peer Evaluation	– 15%

Peer evaluation will be discussed in class closer to the due date, but each group member will evaluate their own performance in the team environment as well as evaluate each other group member's contribution.

**DUE DATE: Thursday, April 18<sup>th</sup>, 8PM.**