Borough of Manhattan Community College
Computer Information Systems
CSC 111 – Introduction to Programming

Assignment 2 – Fall 2020

Due Date: *by Friday October 30, 2020 11:59PM*
How to submit: *upload C++ source files to Blackboard*

In this assignment we will apply the techniques learned in chapters 1 – 6. Your solution must compile, run, and produce the required output. Use the provided CPP file to guide you through the assignment.

Note:

- ✓ ***this is an individual assignment; please do your own work, sharing and/or copying code and/or solution ideas with/from others will result in a grade of 0 and disciplinary actions for all involved parties. If you run into problems and have done your best to solve them, please contact me before/after class or by e-mail.***
- ✓ ***A 20% grade deduction for every day the assignment is late.***

How to submit:

Log into your Blackboard account, click on assignments then Assignment 1. Please upload your source file only (CPP). Your submission must be received by the indicated due date.

Assignment's Instructions

Write a C++ program which performs the following steps. Pay attention to any of the compiler's warning messages. Remember to comment your code. Comments should explain every major step in your code.

1. Display the selection menu shown in Figure 1.
   - ✓ Process the user's selection using a $SWITCH$ structure and ensure to account for lowercase and uppercase entries.
   - ✓ If the user enters A $(or\ a)$, invoke the function $gradeTable(\ \ )$. Once done, prompt the user again.
   - ✓ If the user enters B $(or\ b)$, invoke the function $fileSummary(\ \ )$. Once done, prompt the user again.
   - ✓ If the user enters C $(or\ c)$, terminate the program and print the message $Program\ terminated$.
   - ✓ If the user enters another value, show an error message and PROMPT the user again. The program should never terminate unless the user selects the exit option.
2. $getLetterGrade(\ \ )$: accepts one parameter of type $float$ and returns the corresponding letter grade as follows:

Range		Letter Grade
0	59	F
60	69	D
70	79	C
80	89	B
90	100	A
	Other	X

3. $printGradeTable(\ \ )$: utilizes the function $getLetterGrade(\ \ )$ and prints the table shown in Figure 4. You may pick the proper column width to format and print the table.
4. $fileSummary(\ \ )$:
   - ✓ Open the file "grades.dat".
   - ✓ If the file cannot be opened, display the error message "$Error, cannot\ open\ file: grades.dat$"
   - ✓ Once the file is successfully opened, print the table shown in Figure 6 and use the function $getLetterGrade(\ \ )$.
   - ✓ Compute and print the $minimum, maximum$, and $average$ of all grades in the file. See the bottom of Figure 6

Assessment:

Comments	10
$getLetterGrade(\ \ )$	15
$printGradeTable(\ \ )$ ➢ Formatting ➢ Uses $getLetterGrade(\ \ )$	20
$fileSummary(\ \ )$ ➢ File check ➢ Proper file handle ➢ Format and print all records	30
Menu ➢ Switch (case insensitive and invalid options) ➢ Options ➢ Exit message	25
	100

Figures

```
A) Grade conversion table.
B) Summaries a grade file.
C) Exit.
Select an option():
```

Figure 1: Selection Menu

```
A) Grade conversion table.
B) Summaries a grade file.
C) Exit.
Select an option(): yes

'y' is not a valid choice.

A) Grade conversion table.
B) Summaries a grade file.
C) Exit.
Select an option():
```

Figure 2: invalid entries print an error message and prompts the user again

```
A) Grade conversion table.
B) Summaries a grade file.
C) Exit.
Select an option(): c
Program terminated.
```

Figure 3: option 'c' terminates the program. Note the message before the program ends.

```
A) Grade conversion table.
B) Summaries a grade file.
C) Exit.
Select an option(): a
----------------------
 From   To    Letter
----------------------
    0   59      F
   60   69      D
   70   79      C
   80   89      B
   90  100      A
     Other      X
----------------------
A) Grade conversion table.
B) Summaries a grade file.
C) Exit.
Select an option():
```

Figure 4: option 'A' prints the letter conversion table and prompts the user again

```
A) Grade conversion table.
B) Summaries a grade file.
C) Exit.
Select an option(): b
Error, cannot open file: grades.dat

A) Grade conversion table.
B) Summaries a grade file.
C) Exit.
Select an option():
```

Figure 5: option 'B' prints an error message if the file cannot be opened and prompts the user again

```
A) Grade conversion table.
B) Summaries a grade file.
C) Exit.
Select an option(): b
---------------------------------
     Name        Average   Letter
---------------------------------
  Student01       85.36      B
  Student02       97.22      A
  Student03       54.32      F
  Student04       69.67      D
  Student05       93.76      A
  Student06       88.61      B
  Student07       93.29      A
  Student08       70.63      C
  Student09       73.03      C
  Student10        76.7      C
  Student11       94.17      A
  Student12       46.24      F
  Student13       96.56      A
  Student14       93.77      A
  Student15       73.66      C
  Student16       97.49      A
  Student17       87.05      B
  Student18       68.13      D
  Student19       55.39      F
  Student20       64.26      D
  Student21       99.96      A
  Student22       80.97      B
  Student23        96.4      A
  Student24       92.47      A
  Student25       63.22      D
  Student26       81.38      B
  Student27       91.11      A
  Student28       99.46      A
  Student29       76.49      C
  Student30       68.48      D

    Min: 46.24
    Max: 99.96
Average: 80.97

A) Grade conversion table.
B) Summaries a grade file.
C) Exit.
Select an option():
```

Figure 6: option 'B' reads the file grades.dat, prints the table and prompts the user again