**University of Michigan – Dearborn**

**CIS 150 – Computer Science 1**

**Professor Bacha**

**Lab# 5**

Demetrius Johnson

[meech@umich.edu](mailto:meech@umich.edu)

February 12, 2020

**Table Content**

Contents

[Question 1 3](#_Toc32868799)

[Test Cases 3](#_Toc32868800)

[Source Code – see LAB5Q1.cpp 3](#_Toc32868801)

[Screenshots 3](#_Toc32868802)

[Question 2 6](#_Toc32868803)

[Test Cases 6](#_Toc32868804)

[Source Code – see LAB5Q2.cpp 6](#_Toc32868805)

[Screenshots 6](#_Toc32868806)

[Question 3 9](#_Toc32868807)

[Test Cases 9](#_Toc32868808)

[Source Code – see LAB5Q3.cpp 9](#_Toc32868809)

[Screenshots 9](#_Toc32868810)

[Question 4 12](#_Toc32868811)

[Test Cases 12](#_Toc32868812)

[Source Code – see LAB5Q4.cpp 12](#_Toc32868813)

[Screenshots 12](#_Toc32868814)

# Question 1

## Test Cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test # | Valid / Invalid Data | Description of test | Input Value | Actual Output | Test Pass / Fail |
| 1 | Valid | See if divide by 0 parameter accounted for | totNumStud = 0 | “no students in class…divide by 0 undefined…” | Pass |
| 2 | Valid | See if class average is calculated correctly | totNumStud = 5 | See screenshot of test | Pass |
| 3 | Valid | See if class average is calculated correctly | totNumStud = 9 | See screenshot of test | Pass |
| 4 | Valid | See if class average is calculated correctly | totNumStud = 8 | See screenshot of test | Pass |

## Source Code – see LAB5Q1.cpp

## Screenshots

TEST 1

![A screenshot of a cell phone

Description automatically generated]()

TEST 2

![A screenshot of a cell phone

Description automatically generated]()

TEST 3

![A screenshot of a cell phone

Description automatically generated]()

TEST 4

![A screenshot of a cell phone

Description automatically generated]()

# Question 2

## Test Cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test # | Valid / Invalid Data | Description of test | Input Value | Actual Output | Test Pass / Fail |
| 1 | Valid | See if MAX selection and function works properly | userChoice = ‘A’  firstInt = 4  secondInt = 6 | See screenshot | Pass |
| 2 | Valid | See if MIN selection and function works properly | userChoice = ‘B’  firstInt = 6  secondInt = 7 | See screenshot | Pass |
| 3 | Valid | See if QUIT option works properly to exit the program | userChoice = ‘C’ | See screenshot | Pass |
| 4 | Valid | See if program will re-prompt user for selecting an option within the menu if an unspecified value is entered | userChoice = ‘D’ | See screenshot | Pass |
| 5 | Valid | See if function/program can detect if the two values are equal | userChoice = A  firstInt = 4  secondInt = 4 | See screenshot | Pass |

## Source Code – see LAB5Q2.cpp

## Screenshots

TEST 1

![A screenshot of a computer

Description automatically generated]()

TEST 2

![A screenshot of a cell phone

Description automatically generated]()

TEST 3

![A screenshot of a cell phone

Description automatically generated]()

TEST 4

![A screenshot of a social media post

Description automatically generated]()

TEST 5

![A screenshot of a cell phone

Description automatically generated]()

# Question 3

## Test Cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test # | Valid / Invalid Data | Description of test | Input Value | Actual Output | Test Pass / Fail |
| 1 | Valid | See if winner of two teams function works properly – teamONE as winner | userChoice = ‘A’  see screenshot | see screenshot | Pass |
| 2 | Valid | See if winner of two teams function works properly – teamTWO as winner | userChoice = ‘A’  see screenshot | see screenshot | Pass |
| 3 | Valid | See if winner of two teams function works properly – tie detected | userChoice = ‘A’  see screenshot | see screenshot | Pass |
| 4 | Valid | See if program will re-prompt user for a valid menu option, and if exit option works properly | userChoice = ‘C’  userChoice = ‘B’  see screenshot | see screenshot | Pass |

## Source Code – see LAB5Q3.cpp

## Screenshots

TEST 1

![A screenshot of a cell phone

Description automatically generated]()

TEST 2

![A screenshot of a cell phone

Description automatically generated]()

TEST 3

![A screenshot of a cell phone

Description automatically generated]()

TEST 4

![A screenshot of a cell phone

Description automatically generated]()

# Question 4

## Test Cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test # | Valid / Invalid Data | Description of test | Input Value | Actual Output | Test Pass / Fail |
| 1 | Valid | Check if normal functionality works for calculating min and max for selected number set of input integers | numIntComp = 5  see screenshot | See screenshot | Pass |
| 2 | Valid | Check if program can still detect min and max with all same int values | numIntComp = 11  see screenshot | See screenshot | Pass |
| 3 | Valid | Check functionality for 1 integer selected to be compared | numIntComp = 1 | See screenshot | Pass |
| 4 | Valid | Check program for 0 integers to compared | numIntComp = 0 | See screenshot | Pass |
| 5 | Valid | Check to see if program exits if user inputs a value less than 0 for integers to be compared | numIntComp = -3 | See screenshot | Pass |

## Source Code – see LAB5Q4.cpp

## Screenshots

TEST 1

![A screenshot of a cell phone

Description automatically generated]()

TEST 2

![A screenshot of a cell phone

Description automatically generated]()

TEST 3

![A screenshot of a cell phone

Description automatically generated]()

TEST 4

![A screenshot of a cell phone

Description automatically generated]()

TEST 5

![A screenshot of a cell phone

Description automatically generated]()