

**FACULTY OF ENGINEERING** SCHOOL OF COMPUTING

**SUBJECT:**

SECJ 1013 - 05 (PROGRAMMING TECHNIQUE I)

SEMESTER 1, 2020/2021

**ASSIGNMENT 2**

**GROUP MEMBERS:**

|  |  |  |
| --- | --- | --- |
| NAME | MATRIK NO. | PHONE NO. |
| 1. Lee Jia Xian | A20EC0200 | 011-20621573 |
| 2. Shingthai Srisoi | A20EC0147 | 014-3639822 |
| 3. Tan Yong Sheng | A20EC0157 | 019-9839702 |

**SUBMIT TO:**

DR. GOH ENG SU

1. There are five operators shown in the expression as in the given figure. Label the order of execution for each operator in the boxes as stated in the expression. The operator that will be executed first should be labeled as 1, the second operator to be executed should be labeled as 2, and so on. Finally give the result of the expression according to this sequence of executions.

[3 marks]

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **z** | **=** | **(** | **12** | **+** | **4** | **)** | **/** | **4** | **+** | **30** | **/** | **3** | **-** | **3** |
|  | | | | 1 |  | | 2 |  | 4 |  | 3 |  | 5 |  |

# Answer: 11

1. What is the output for the following statements marked (a) to (e). Write your answer in the space provided according to the sequence required.

[5 marks]

**double val = 10.34567;**

**cout << setprecision(6) << val << endl; // (a)**

**cout << static\_cast<int>(val)/2 << endl; // (b) cout << "\t" << setprecision(3) << val << ", " ; // (c) cout << setw(6) << val\*5 << endl << endl; // (d) cout << showpoint << fixed << setw(9) << val << endl; // (e)**

# Answer:

10.3457

5

10.3, 51.7

10.346

1. Refer to **Program A.1** below which is incomplete. As a result of the prompt on line **9**, assume that the input string entered is "ABCDEFG HIJKL". Write the corresponding output to be displayed for each of the corresponding codes in Code 1, Code 2 and Code 3 as given in the boxes after **Program 1**.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | E | F | G |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| A | B | C | D | E | F | G |  | H | I | J | K |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | A | B | C | D | E | F | G |  |

[7 marks]

|  |  |  |
| --- | --- | --- |
| **1**  **2**  **3**  **4**  **5**  **6**  **7**  **8**  **9**  **10**  **11**  **12**  **13**  **14**  **15** | **// Program 1 #include <iostream> #include <iomanip> using namespace std;**  **int main( )**  **{**  **char x[13];**  **cout << " Enter a string : ";**  **// either Code 1, 2 or 3**  **// will be placed here**  **return 0;**  **}** | |
| **Code 1: [ 2 marks ]**  **cin>>x; cout<<x<<endl;** | | Output: |
| **Code 2: [ 2 marks ]**  **cin.getline(x,13); cout<<x<<endl;** | | Output: |
| **Code 3: [ 3 marks ] cin>>x;**  **cout<<setw(13)<<x<<endl;** | | Output: |

1. The following C++ program, **Program 2** cannot be compiled. Rearrange the lines in the correct sequence in order for the program to be compiled and executed to produce the output as indicated.

[5 marks]

|  |  |
| --- | --- |
| **1** | **// Program 2** |
| **2** | **cout << "Success\n";** |
| **3** | **cout << " Success\n\n";** |
| **4** | **int main()** |
| **5** | **cout << "Success"; }** |
| **6** | **using namespace std;** |
| **7** | **#include <iostream>** |
| **8** | **cout << "Success\n";** |
| **9** | **{** |
| **10** | **return 0;** |

# Program output :

**Success Success**

**Success Success**

Use the table below to rearrange the lines based on the output given.

|  |  |
| --- | --- |
| **1** | // Program 2 |
| **2** | #include <iostream> |
| **3** | using namespace std; |
| **4** | int main () |
| **5** | { |
| **6** | cout << “Success\n” ; |
| **7** | cout << “Success\n\n”; |
| **8** | cout << “Success”; |
| **9** | cout << “Success\n”; |
| **10** | return 0;} |

1. Write the corresponding conditional expression for the following if else statements.

[12 marks]

|  |  |  |
| --- | --- | --- |
|  | if…else statement | Conditional expression |
| i. | **if (score >= 50)**  **{ numPass++; cout<<“Pass”; }**  **else**  **{ numFail++;**  **cout<<”Please try again.”; }** | score >= 50 ?( numPass++, cout << “Pass”) : (numFail++, cout << “Please try again.”); |
| ii. | **if (cpa >= 2.0)**  **{ if (cpa >= 3.5)**  **status = “Dean’s List”; else**  **status = “Normal Pass”; }**  **else**  **{ if (cpa >= 1.7)**  **status = “Probation”; else**  **status = “Fail”; }** | cpa >= 2.0 ? ((cpa >= 3.5)? status = “Dean’s List”: status = “Normal Pass”) :(( cpa >= 1.7) ? status = “Probation” : status = “Fail”); |

1. Write C++ **if** statement code fragments to satisfy the given conditions.

[10 marks]

|  |  |
| --- | --- |
| i. | Check the range of frequency, **freq** to be between 100Hz and 10000Hz. Display "**Acceptable**" if within the range and "**Unacceptable**" if not.  (3 marks) |
|  | if (freq > 100 && freq <10000 )  cout << "Acceptable";  else  cout << "Unacceptable"; |

|  |  |
| --- | --- |
| ii. | Check the prerequisite for a soldier candidate to be of **age** between 18 to 30 years, **weight** between 50 to 65kg and **height** must be greater than 156m. Display "**Fulfill requirements**" or "**Do not fulfill requirements**" based on these conditions.  (3 marks) |
|  | if(age>18&&age<30 && weight>50&&weight<65 && height>156){  cout<<” Fulfill requirements"”;  }else{  Cout<<"Do not fulfill requirements";  } |
| iii. | Henry wants to buy a car. It must be under one of these conditions. Either:   1. The **year** made: after 2010, cylinder capability: **cc** between 1.5 to 2.0.   ***or***   1. The **year** made: before 2010 , cylinder capability: **cc** greater than 2.0.   His decision either to "**Purchase car**" or "**Do not purchase car**" should be reflected in the code.  (4 marks) |
|  | if (year>2010 && cc>1.5&&cc<2.0 || year<2010 && cc>2.0){  cout<<”Purchase car”;  }else{  cout<<”Do not purchase car”;  } |

1. What is the output for the following code excerpts?

[5 marks]

|  |  |
| --- | --- |
| **Code** | **Output** |
| **int n = 0; if (n = 0)**  **cout << "Yes"; else**  **cout << "No";**  **}** | No |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **int i=10, j=3,k = 20; cout << ((j < 4 )||** | | | **(j** | **==** | **5)** | **&&** | **(i** | **<=** | **k));** | 1 |
| **int x =** | **13, y**  **y)**  **> 0)**  **= x \***  **if (y**  **x =**  **x;** | **= 9;** | | | | | | | | 117 |
| **if (x >=** |  | | | | | | | |
| **if (y** |  | | | | | | | |
| **x** | **y;** | | | | | | | |
| **else** | **< 4)** | | | | | | | |
|  | **x - y;** | | | | | | | |
| **cout <<** |  | | | | | | | |

1. Fill in the spaces provided in order for the program segment to produce the output as shown.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **// Question 9.a. – [8 marks]**  **int x =15;**  **do {**  **x--;**  **if (x %2 == 1)**  **continue;**  **cout << x << " ";**  **} while (x >=4 );** | | | | |  | **//(a)**  **// (b)**  **// (d)** | **and** | **(c)** |
| ***Output:***  **14 12** | **10** | **8** | **6** | **4** | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **// Question 9.b. – [6 marks] int y = 2;**  **do {**  **if (y>300)**  **break;**  **cout << y << " ";**  **y = pow(y, 2);**  **} while (y >= 2** | | **);** | **//**  **//**  **//** | **(a)**  **(b)**  **(c)** |
| ***Output:***  **2 4 16** | **256** | | | |

1. Based on the flowchart given in **Figure 1**, answer parts (i) to (iii) of this question.

[10 marks]

Start

sum = 0

i = 5

i < 10?

false

true

sum = sum + 1

i = i + 2

Display sum

End

# Figure 1

* 1. Convert the given flowchart into its equivalent C++ code excerpt. (4 marks)

#include <iostream>

using namespace std;

int main ()

{

int sum = 0, i = 5;

while (i < 10)

{

sum += 1;

i += 2;

}

cout << sum << endl;

return 0;

}

* 1. How many times the loop repeat. (1 mark)

3

* 1. Modify your code by using decrement counter loop without changing the variables involved and the number of loops involved. (4 marks)

#include <iostream>

using namespace std;

int main ()

{

int sum = 0, i = 5;

while (i > 0)

{

sum += 1;

i -= 2;

}

cout << sum << endl;

return 0;

}

1. **Program 3** is able to count the number of input character of **A**, **B** and **C**. The program will loop reading the input until the sentinel value **e** is being input. The sample output of the program is as shown:

[20 marks]

**Enter the letter grades[Enter 'e' character**

**A**

**Enter the letter grades[Enter 'e' character**

**B**

**Enter the letter grades[Enter 'e' character**

**C**

**Enter the letter grades[Enter 'e' character**

**C**

**Enter the letter grades[Enter 'e' character**

**C**

**Enter the letter grades[Enter 'e' character**

**e**

**Totals for each letter grade are: A: 1**

**B: 1**

**C: 3**

**Press any key to continue . . .**

**to end input]**

**to end input] to end input] to end input] to end input]**

**to end input]**

Complete **Program 3** based on the comments given

|  |  |
| --- | --- |
| 1 | *//****Program 3*** |
| 2 | #include <iostream> |
| 3 | using namespace std; |
| 4 |  |
| 5 | int main() |
| 6 | { |
| 7 | char grade; ***// one grade*** |
| 8 | int aCount = 0; ***// number of characterAs*** |
| 9 | int bCount = 0; ***// number of Bs*** |
| 10 | int cCount = 0; ***// number of Cs*** |
| 11 |  |
| 12 | cout << "Enter the letter grades[Enter 'e' character to end |
| 13 | input]"<< endl; |
| 14 | cin>>grade; |
| 15 |  |
| 16 | ***// loop: as long as sentinel value has not been achieved*** |
| 17 | while(grade!=’e’){ *// (a) - 2 marks* |
| 18 |  |
| 19 |  |

20 ***//write appropriate statement for testing input cases***

21 switch(grade) { *// (b) - 2 marks*

22

23 ***//in case of input A, increment variable aCount*** *- 3 marks*

24

25 case ‘A’ : *// (c)*

26 aCount++; *// (d)*

27 break; *// (e)*

28

29 ***//in case of input B, increment variable bCount*** *- 3 marks*

30

31 case ‘b’: *// (f)*

32 bCount++; *// (g)*

33 break; *// (h)*

34

35 ***//in case of input C, increment variable cCount*** *- 3 marks*

36

37 case ‘C’: *// (i)*

38 cCount++; *// (j)*

39 break; *// (k)*

40

1. ***//add the statement to catch all other alphabets and prints***
2. ***//“Incorrect letter grade entered.”*** *- 3 marks*

43

44 default : *// (l)*

45 cout << “Incorrect letter grade entered.\n”; *// (m)*

46 break; *// (n)*

47

48 } ***// end test cases***

49

50 ***//ask for another input letter grades*** *- 2 marks*

51 cout << "Enter the letter grades[Enter 'e' character to end input]"<< endl;*// (o)*

52 cin>>grade; *// (p)*

53

54 } ***// end loop***

55

56 ***// output summary of results*** *- 2 marks*

57

58 cout<<”A: ”<<aCount<<endl; *//* ***(q) display number of A grades***

59 cout<<”B: ”<<bCount<<endl; *//* ***(r) display number of B grades***

60 cout<<”C: ”<<cCount<<endl; *// (****s) display number of C grades***

61

1. return 0;
2. } ***// end function main***
3. Write a **Program 4** that allows user to:
4. Input integer number(s) continuously until they decide to terminate/quit the program by entering ‘0’ (zero).
5. Count how many times the user input any extreme values that are numbers, which **not in the range** of 30 to 200.
6. Calculate the average value for the user inputs.

**Figure 2** shows the output sample for the case of extreme values are being input, while

**Figure 3** shows the output sample for the case of no extreme values are being input.

**Enter a number (0 to quit) : 201 Enter a number (0 to quit) : 205 Enter a number (0 to quit) : 29 Enter a number (0 to quit) : 25 Enter a number (0 to quit) : 0**

**Average is: 115.00**

**There were 4 extreme values Press any key to continue . . .**

**Enter a number (0 to quit) : 30 Enter a number (0 to quit) : 200 Enter a number (0 to quit) : 40 Enter a number (0 to quit) : 0**

**Average is: 90.00**

**There were 0 extreme values Press any key to continue . . .**

Figure 2 Figure 3

Complete **Program 4** below based on the given instructions within the program.

[10 marks]

|  |  |
| --- | --- |
| 1 | ***//Program 4*** |
| 2 | ***//Include suitable libraries @ header file*** (2 marks) |
| 3 | **#include <iostream>** |
| 4 | **#include <iomanip>** |
| 5 | **using namespace std;** |
| 6 |  |
| 7 | **#define LOW 30 *// lowest value in range*** |
| 8 | **#define HIGH 200 *// highest value in range*** |
| 9 | **#define EXIT 0 *// sentinel value*** |
| 10 |  |
| 11 | **int main()** |
| 12 | **{** |
| 13 | **int userVal; *// user input*** |
| 14 | **int extremeCnt = 0; *// extreme values counter*** |
| 15 | **int sum = 0; *// total values of user inputs*** |
| 16 | **int num = 0; *// no of user inputs*** |
| 17 | **double avg; *// average of user inputs*** |
| 18 |  |
| 19 | ***// Prompt user to input the first number*** |
| 20 | **cout << "Enter a number ("<< 0 <<" to quit):";** |
| 21 | **cin >> userVal;** |
| 22 |  |

1. ***/\* Write WHILE loop that allow user to:***
2. ***- Input number(s) continuously until user enter ‘0’***
3. ***- Count no of user inputs and extreme values counter***
4. ***- Calculate total value of user inputs***
5. (4 marks) ***\*/***

28

29

30 while(!(userVal == 0))

31 {

32 num++;

33 if(userVal<LOW || userVal>HIGH)

34 {

35 extremeCnt++;

36 }

37 sum += userVal;

38 cout << "Enter a number (" << 0 <<" to quit)";

39 cin >> userVal;

40 }

41

1. ***/\* Calculate the average value for the user inputs***
2. ***Display the number of extreme values*** (4 marks) **\*/**

44

45 avg = sum/num;

46 cout << "Average is: " << fixed << setprecesion(2) << avg << endl;

47 cout << "There were " << extremeCnt << "extreme values";

48 system(“pause”);

49

50 **return 0;**

51

52 **} *//end main***