SAMPLE Q&A

PART B

Q1
a)Difference between binary and unary operator.

Ans:

UNARY OPERATORS	BINARY OPERATORS
A **unary operator** is one that operates upon a single operand/argument and performs an operation.	Binary operators are those operators that work with two operands.
EXAMPLES: Lets x is a variable(Operand). Pre Increment: ++x Post Increment: x++ ● Decrement: −−x , x−− ● Address: &x. ● Indirection: *x. ● Positive: +x. ● Negative: −x. ● Ones' complement: ~x. ● Logical negation: !x.	**EXAMPLES:** **+,-,*,/ etc…** A common **binary** expression would be a + b The addition **operator** (+) surrounded by two operands.

b) Explain logic error and syntax error.

Ans: There are generally two types of errors in programing:
1. syntax errors
2. logic errors
Syntax error:
Syntax error is equivalent with grammatical error in ENGLISH language. If some words, statements are unknown to the compiler, the compiler produces an error which is known as Syntax error..
Logic error:
Logic errors occur when a program does not do what the programmer expects it to do.

c) State two ways to get input data from a program.

Ans:
- std::cin
- std::getline

##

2 a-i) DEclare a variable named discount and assign 20% to it
Ans:
double discount = 0.2;

ii) Declare a constant named UNI
Ans:
const string UNI = "UiTM";

2 b-i)

```
int main() {
  int x = 5 * 3 - 9 / 2 % 2 + 6;
  cout << x;
  return 0;
}
```
OUTPUT: 21

STEPS:
5 * 3 - 9 / 2 % 2 + 6
15 - 9 / 2 % 2 + 6
15 - 4   % 2 + 6
15 - 0 + 6
15   +   6
21

ii) 42 % 5 +7 -(2+8)/ 3
OUTPUT: 6

Explanation:

42 % 5 +7 -(2+8)/ 3

42 % 5 +7 -10/ 3

2 +7 -10/ 3

2 +7 - 3

9 - 3

6

PROGRAM:
```
int main() {

  int x =  42 % 5 +7 -(2+8)/ 3;
  cout << x;
  return 0;
}
```

QUESTION 3
3-a-i)
Ref: jun2019_3ai.cpp
jun2019_3ai.cpp

a-ii) jun2019_3aii.cpp
jun2019_3aii.cpp

OUTPUT:
5
9
13
17

b)Answer the following questions

i)if the input is 10664, trace the output of the following program.

```cpp
int main()
{

int num,A,B,C,D,E,rem;
cout << "enter a five digit number:";
cin >> num;                              // num = 10664
A = num/10000;                      //  A =  1
rem = num%10000;                   //   rem = 664
B = rem/1000;                        // B = 0
rem = rem%1000;                     // rem = 664
C = rem/100;                         // C = 6
rem = rem%100;                      // rem = 64
D = rem/10;                          // D = 6
rem =rem%10;                        // rem = 4
E = rem;                             // E = 4
cout <<A<<" "<<B<<" \n"<<C<<"\n"<<D<<" "<<E <<endl;
  if ((A==E)&& (B==D))                        // 1 == 4 is false, so else executess
   cout <<"\n PALINDROME NUMBER";
   else
     cout <<"\n NOT PALINDROME NUMBER ";
return 0;

}
```

output
1 0 6 6 4 NOT PALINDROME NUMBER

ii)
Refer: jun2019_3bii.cpp
jun2019_3bii.cpp

4-a i)
>   **Jun2019_4ai**
>   jun2019_4ai.cpp

```cpp
/* *********************
* DISPLAY ALL THE NUMBERS RANGING FROM 30 to 150 THAT ARE DIVISIBLE BY 5 and
7
* *********************/

#include <iostream>

using namespace std;


int main() {
 for(int num = 30; num <= 150; ++num) {
  if((num % 5 == 0) && (num % 7 == 0)) {
    cout << num << " ";
  }
 }
 return 0;
}

        // One more way
// If a number is divisible be 5 & 7, it means it should be divisible by 5*7 = 35
/*int main() {
 for(int num = 30; num <= 150; ++num) {
  if((num % 35 == 0)) {
```

```
    cout << num << " ";
  }
 }
 return 0;
}
*/
```

4 a-ii) jun2019_4aii.cpp
jun2019_4aii.cpp

Factorial of 1 = 1
Factorial of 2 = 2
Factorial of 3 = 6
Factorial of 4 = 24

4 b-i)
Refer attached program : **jun2019_4bi.cpp**
jun2019_4bi.cpp

ii)
Refer attached program: **jun2019_4bii.cpp**
jun2019_4bii.cpp

5. WRITE FUNCTION HEADER
```
 //a-i)
 /**
 * @brief Calculate and print the total price
 * @param quantity: quantity of the item
 * @param price: price of each item
 */
void calcPrice(int quantity, float price)

 //ii)
/**
 * @brief calcDiscount calculates total price after discount
 * @param totalPrice: total price of item
 * @return total price after discount
 */
float calcDiscount(float totalPrice)
```

//iii)

```
/**
* @brief calcTax calculates and returns total tax as reference param
* @param totalPrice: total price of item
* @param totalTax: Out param to return total tax.
*/
void calcTax(const float totalPrice, float& totalTax)
```

ALTERNATE AND BEST ANSWERS IF YOU CAN UNDERSTAND:
i) void calcPrice(const int& quantity, const float& price)
ii) float calcDiscount(const float& totalPrice)
iii) same as above :
void calcTax(const float totalPrice, float& totalTax)

##
5-b)
  Jun2019_5b.cpp
jun2019_5b.cpp

#########################
##########################
PART C

QUESTION-1:
Reference:
part_c_Q1.cpp

QUESTION-2:
Reference:
part_c_q_2.cpp