Question 1:   (8 Marks)

A. **(4 Marks)** Determine the output for each of the following code snippets (assuming successful compilation):

a) (2 Mark)	b) (2 Mark)

```# include <iostream>```
```void draw (void) ;```
```void main ( ) {```
```draw ( );```
```draw ( );```
```cout << "Welcome "<< endl;```
```draw ( ) ;```
```cout << " First Year " ;```
```draw ( ) ;```
```}```
```void draw (void) {```
```for (int i = 10 ; i < 14 ; i ++) {```
```cout << " * " ; }```
```cout << endl ; }```
	```for (int i = 1; i <= 10; i++) {```
	```if (i == 7 || i == 8 || i == 9) {```
	```continue;```
	```}```
	```cout<<i;```
	```}```

Answer:
   * * * *
   * * * *
  Welcome
  * * * *
  First Year * * * *
	Answer:
	
	1 2 3 4 5 6 10

B. **(4 Marks)** Complete the following sentences:
1. Any C++ program has at least **one** function(s).
2. If Z= 6, the value of the following expression: sqrt (Z + 30) is **6**.
3. The value of the following expression: power (4, 2) is **16**.
4. The basic mathematical functions can be used in any C++ program by including a library called **cmath**.

Question 2: (10 Marks)

A. (5 Marks) Write a complete program that reads the parameters of a quadratic equation and prints its real or imaginary roots.

Hint: $x_{1,2} = (-b \pm \sqrt{b^2 - 4.a.c}) / (2.a)$

Examples:

Input:	a=1, b=5 and c= 6	a=1, b= -4 and c=4	a=1, b= 2 and c=2
Output:	X_1= -2 and X_2= -3	$x_1 = x_2 = 2$	X_1= -1+1i and X_2= -1-1i

First, you should calculate the Discriminant: $D = \boldsymbol{b^2 - 4.a.c}$, then:

- If the discriminant is positive, then there are two distinct roots
  $(-b + \sqrt{D}) / (2.a)$ and $(-b - \sqrt{D}) / (2.a)$
- If the discriminant is zero, then there is exactly one real root
  $-b/(2.a)$
- If the discriminant is negative, there are two distinct complex roots
  $(-b/(2.a) + i \ \sqrt{-D})/2a)$ and $(-b/(2.a) - i \ \sqrt{-D})/2a)$

Answer:

```cpp
#include <iostream>
#include <cmath>
using namespace std;
intmain() {
double a, b, c, d;
cout<< "Enter a, b, and c: ";
cin>> a >> b >> c;
d = b * b - 4 * a*c;
if (d > 0) {
cout<< "x1 = " << (-b + sqrt(d)) / (2 * a)
<< ", x2 = " << (-b - sqrt(d)) / (2 * a);
} else if (d < 0)
cout<< "x1,2 = " << -b / (2 * a)
<< " + - " << sqrt(-d) / (2 * a) << "i";
else
cout<< "x1,2 = " << -b / (2 * a);
return 0;
}
```

B. (5 Marks) Write a complete C++ program that accepts a character grade (A, B, C or F) from the user and prints its equivalent integer GPA (4, 3, 2 and 0 respectively) using <u>switch case.</u>
Answer:

```
#include <iostream>
int main( ) {

char Grade;
cout << "Please enter your grade:" << endl ;
cin >> Grade ;

switch (Grade){
case 'A':
cout << "your GPA is 4";
break;
case 'B':
cout << "your GPA is 3";
break;
case 'C':
cout << "your GPA is 2";
break;
case 'F':
cout << "your GPA is 0";
break;
default:
cout << "Wrong grade, you must enter a grade in the range from 0 to 100";
}
        return 0;
}
```

Question 3:

A. **(6 Marks)** Write a complete C++ program that repeats printing a specific character according to a specific number. The program should ask the user to enter a character and a number. The program should then display the character many times as the number. [For example, if the user enters "$" and "5", the program should display "$$$$$"].

B. **(4 Marks)** Write **for** loops that will print the following patterns:

a) (2 Mark)	b) (2 Mark)
$ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $ $	1 1 2 1 2 3 1 2 3 4

Answer:
for (int i = 1; i <= 4; i++) {
for (int j = 1; j <= 5; j++) {
cout <<" $ ";
}
cout<< "endl";
}
	Answer:
	for (int i = 1; i <= 4; i++) {
	for (int j = 1; j <= i; j++) {
	cout << j;
	}
	cout<< "endl";
	}

Question 4:   (12 Marks)

A. (6 Marks) Write a full program that reads an array of *n* numbers and prints its range. The range of an array is the difference between its maximum and minimum values. The program should consist of two functions:

(i) "min", for finding the minimum value, and
(ii) "max", for finding the maximum value.

Solution:

```
#include <iostream>
using namespace std;
//Function Prototype
void read(float[], int);
float max(float[], int);
float min(float[], int);
int main() {
const int N = 100;
int n;
float x[N];
//Read array size
do {
cout << "Enter n [1," << N << "]: ";
cin>> n;
if (n < 1) {
cout << "Too small!" <<endl;
}
if (n > N) {
cout << "Too large!" <<endl;
}
} while (n < 1 || n > N);
read(x, n);
cout<< "The range is: " <<max(x, n) - min(x, n) <<endl;
return 0;
}
//Function Definition
//a)
void read(float x[], int n) {
for (int i = 0; i< n; i++) {
cout<< "Enter element number " <<i<< ": ";
cin>> x[i];
}
}
//b)
float min(float x[], int n) {
float minX = x[0];
for (int i = 1; i< n; i++)
if (minx >  x[i])
minX = x[i];
return minX;
}
//c)
float max(float x[], int n) {
float maxX = x[0];
for (int i = 1; i< n; i++)
if (maxX <  x[i])
maxX = x[i];
return maxX;
}
```

B. **(6 Marks)** Write a full program that reads a 6*6 array from the user. The program should then calculate and display:
- The summation of all elements in the array,
- The summation of the diagonal elements only,
- The summation of the opposite diagonal only.

Solution:

```
# include <iostream.h>
void main ( ) {
int  X [6] [6] ;
int total = 0 ;
intsumdiagonal = 0;
intOdiagonal;
for ( inti = 0 ; i< 6 ; i ++ ) {
    for ( int j = 0 ; j < 6 ; j ++ ) {
cin>> X [ i ] [ j ] ;  }  }

// A.
for ( inti = 0 ; i< 6 ; i ++ ) {
    for ( int j = 0 ; j < 6 ; j ++ ) {
    total = total  +  X [ i ] [ j ] ;  }   }
cout<<" The summation of the array elements is " << total <<endl ;
}
// B.
for ( inti = 0 ; i< 6 ; i ++ ) {
    for ( int j = 0 ; j < 6 ; j ++ ) {
if  (i = = j ) {
sumdiagonal  =sumdiagonal  +  X [ i ] [ j ] ;
}   }   }
cout<<" The summation of the diagonal elements is " <<sumdiagonal ;
//C.
for ( inti = 0 ; i< 6 ; i ++ ) {
    for ( int j = 0 ; j < 6 ; j ++ ) {
if  (i + j == 5 ) {
Odiagonal  =Odiagonal  +  X [ i ] [ j ] ;
}   }   }
cout<<" The summation of the diagonal elements is " <<sumdiagonal ;
}
```

Good Luck
Shady Yehia Elmashad