Name: .....

1. [3 pts] Fill the following table.

a = 2, b = 5, c = 6;	output
b = -- a; cout << a << " " << b << endl;	
c = a * b++; cout << " " << b << " " << c;	
bool x = (-- c > 5); cout << c << " " << x ;	

2. [2 pts] How many times will the following loop display "Test 3"?

```
for (int i = 0; i <= 20; i++)
   cout << "Test 3" << endl;
```

.....

3. [2 pts] How many numbers will the following loop display?

```
int i = 1;
while (i < 50)
{
  cout << i << " ";
  i *= i;
}
```

.....

4. [2 pts] What is *the last number* displayed out when the following fragment executes?

```
for (int i = 1; i <= 5; i++)
{
        cout << i * (i + 1) << endl;
        cout << endl;
}
```

.....

5. [2 pts] How many times will the following loop display COSC?

```
for (int i = 0; i <= 6; i++)
{
        for (int j = 0; j <= 4; j++)
        {
                cout << "COSC"<< endl;
        }
        cout << endl;
}
```

.....

6. [2 pts] What is the output of the following code

```
for (int j = 7; j > 4; j--)
   cout << j << " ";
```

.....

.....

.....

.....

7. [2 pts] What is the output of the following code

```
int i = 1;
for( ; i <= 10; i += 2)
    cout << i << endl;
```

.....

.....

.....

.....

.....

8. [4 pts] What would the following code display on the screen.

```
int sum,
    num;
for (sum = 10, num = 0; sum <= 30; num = num + 2)
  sum = sum + num;
cout << num << " " << sum << endl;
```

.....

.....

.....

.....

.....

.....

.....

9. [3 pts] What would the following code display on the screen.

```
for (int i = 1; i <= 2; i++)
{
        for (int j = 2; j <= 5; j++)
        {
                cout << i << " " << j << endl;
        }
        cout << endl;
}
```

.....

.....

.....

.....

.....

.....

.....

.....

.....

10. [4 pts] What would the following code display on the screen.

```
int a = 2;

while ( a < 8 )
{
        for (int b = 2; b <= 6; b++)
        {
                cout << a << b << "\t";
                a++;
        }
        cout << endl;
}
```

11. [4 pts] What would the following code display on the screen.

```
for (int j = 4, k = 7; j <= k; j += 2, k++)
   cout << j << " " << k << endl;
```

12. [4 pts] What would the following code display on the screen.

```
for (int i = 1; i <= 5; i++)
{
        for (int j = 1; j <= i; j++)
        {
                cout << j << " ";
        }
        cout << endl;
}
```

13. [4 pt]

```
int num = 20, count = 1;
while (num >= 10 && num <= 50)
{
   num += 10;
   count++;
}
cout << count << " " << num << endl;
```

14. [5 pts]

```
int num = 50, count = 0, sum = 100;
do
{
   num += 50;
   sum += num;
   count++;
}
while (sum <= 500);

cout << count << num << sum << endl;
```

15. [5 pts] Write a code segment that outputs the following; **Define all the variables you would use.**
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

16. [5 pts] Write a code SEGMENT that generates a list of menu options (see the SAMPLE output given below) and take a user input. Your program should make sure that the user input is a valid menu choice. When the input is invalid your program should re-display the menu until user enters a valid choice. **Define all the variables you would use.**

```
Choose and operation
1. Find the total area of all the faces
2. Find the volume
3. Find the total lengths of sides
4
Invalid choice.

Choose and operation
1. Find the total area of all the faces
2. Find the volume
3. Find the total lengths of sides
-1
Invalid choice.

Choose and operation
1. Find the total area of all the faces
2. Find the volume
3. Find the total lengths of sides
2
```

17. [10 pts] Write a for loop to compute the sum of squares of first 20 POSITIVE integers and output the sum: i.e., $1^2 + 2^2 + 3^2 + \ldots + 20^2$ **. Define all the variables you would use.**

18. [10 pts] Use loops to output the following numbers with the same formatting. You should not use hard coded numbers to get the output. **Define all the variables you would use.**

(10: 50) (10: 51) (10:52) (10:53) (10:54) (10:55)
(11: 50) (11: 51) (11:52) (11:53) (11:54) (11:55)
(12: 50) (12: 51) (12:52) (12:53) (12:54) (12:55)
(13: 50) (13: 51) (13:52) (13:53) (13:54) (13:55)

19. [5 pts] Write a code segment that outputs 10 non-negative random numbers in decreasing order using a loop (You DO NOT need to write the preprocessor directives or seeding). The program should output only one number at each iteration. The first random number should be below 100. **Define all the variables you would use.**

20. [5 pts] Write a loop that interactively reads test score values until a test score value of -1 is entered. It also computes and displays the sum of the test scores. **Define all the variables you would use.**

21. [15 pts] Write a FULL program that reads a list of numbers from a file named: "*temperature.txt*" and outputs the list of numbers, in the same order, followed by the maximum of those on the monitor.