Lab 4: Functions and Arrays

Question 1

int main(){

int a = 5; // declares the variable a with value 5

test1(a); // increments the value of a to 6 but it stays in test1 as it is a local variable

cout << a; // prints 5 as the local value remains the same

test2(a); // sends the pass-by reference of “a”, and modify the original variable to 6,cand prints the global variable x value which is 0

cout << a; // prints the value 6, coz the test2 modify the original variable

test3(); // prints 2, the value of x and then increments it but do not do anything

test4(); // prints 2, the value of x and then increments, but this time it increases the value and uses it as it a static variable

test3(); // prints 2, the value of x and then increments it but do not do anything

test4(); // prints 3, from the previous call, and then increments it

return 0;

}

The final output is: 5062223

Question 2

1. True, because passing by reference avoids copying large data, it's faster and uses less memory.
2. True because when a parameter is passed **by** reference, any changes made to it inside the function will affect the original variable
3. True, because passing by value means a copy of the argument is passed to the function. So even if the function changes the parameter, it only changes the copy
4. False because the function will only change the copy but not the original variable

Question 3

```
#include<iostream>
Using namespace std;
Bool isTrue(int num){
static int previous[100];
static int count = 0;
for(int i =0; i < count; i++){
if(previous[i] == num){
return false;
}
}
previous[count++] = num;
return true;
}
```

Question 4

Cout << x[0] print 1 because at index 0 1 is stored

Cout << x prints the memory address which is 100

Cout << x[3} is out of bound but this is undefined and since y is available right x[2], it will print 8

Question 5

The output is as follows:

10

2

1

2

3

1

0

0

0

0

0

0

The main function calculates the number of elements in an array

In function f(), it calculates the size of a pointer, not the array

X[3]++ increments the a[3] from 0 to 1, then printing the array

Question 6

The first error is int arr[n], we can not use n if it is not declared as a constant before, it should be const int n = 10;(Syntax error)

Int arr[n]

The second error is in loop condition because we are trying to access the elemnst which are out of bound (i <= n) it should be for (int = 0; i<n;i++) (Run time error)

The third error is using sqrt without importing the cmath library (Syntax error)

The fourth error is mismatch assignment we are assigning sqrt whihc returns a double to an int array , we should declare array as double arr[n] (Logic error)