Homework 6 Solutions

1.Ans) n=pow(x,3).

2.Ans) (2*xi+y/xi)/4

3.Ans) n(n+1)/2

4.Ans) It prints the binary equivalent of the input integer.

5.Ans) The function recur() calculates and returns ((1 + 2 … + x-1 + x) +y) which is x(x+1)/2 + y. Thus the output would by 6*7/2+3=24

6.Ans) 7

Explanation:- After executing the cout statement in main, function myFunc1 is called for the first time. Within this (first) invocation of myFunc1, the first character 'a' in the input is read. Since this is not 'X', myFunc1 is called again (this time recursively).

Within this (second) invocation of myFunc1, the next character 'b' in the input is read. Once again, this is not 'X', and so myFunc1 is called recursively again.Within this (third) invocation of myFunc1, the next character 'X' in the input is read. This time, myFunc1 is not called recursively, and the instruction after "if (c != 'X') {myFunc1();}" is executed. Therefore, the last character read in this invocation of myFunc1 (i.e. 'X') is output, and this (third) invocation of myFunc1 returns.

We now return to the second invocation of myFunc1, and the next instruction after "if (c != 'X') {myFunc1();}" is executed. Therefore the character that was last read in this (second) invocation of myFunc1 (i.e. 'b') is output, and this (second) invocation of myFunc1 returns.

We now return to the first invocation of myFunc1, and similarly execute the instruction after "if (c != ' ') {myFunc1();}". The character that was last read in this (first) invocation of myFunc1 (i.e. 'a') is output, and this (first) invocation of myFunc1 returns.

We are now done with the first invocation of myFunc1 from within main. The next call (overall, fourth) of myFunc1 from within main is now made. By following the same reasoning as above,

you will find that myFunc1 is invoked an additional 3 times (until the next 'X' is read from the input). These invocations of myFunc1 will output the characters in the order 'X' (output from the seventh invocation of myFunc1), 'a'(output from the sixth invocation of myFunc1), 'a' (output from the fifth invocation of myFunc1), and 'b' (output from the fourth invocation of myFunc1).

The main program ends at this stage, ignoring remaining characters in the input.

7.Ans) The highest power of 2 divisible by x

8.Ans)

```
#include<simplecpp>

int result = 0, i = 0;
int mult_fun(int m,int n){
	if(i < m){
		result = result + n;
		i++;
		mult_fun(m,n);
	}
	return result;
}

main_program{
	int a =0 , b=0;
	cin >> a >> b;
	mult_fun(a, b);
	cout<< result <<endl;
}
```

9.Ans)

```
#include<simplecpp>
int factorial(int n)
{
	if(n == 0 || n == 1) return 1;
	return (n*factorial(n-1));
}


main_program
```

```
{
        int n;
        cout<< "Enter no. : ";
        cin>>n;
        cout << factorial(n);
}
```

10. Ans)

```
int gcd(int x, int y) {
    if ( y == 0 )
        return x;
    else if ( x >= y && y > 0 )
        return gcd(y, x%y);
}
```

11.Ans)

```
bool isPalin(int arr[], int start, int end){
        if(end - start < 2) return true;
        else if(arr[start] != arr[end-1] ) return false;
        else return isPalin(arr, start+1, end-1);
}
```

12.Ans) ABCDE

Each time we are assigning 65 + i. In first iteration i = 0 and 65 (A) is assigned to the first element of array. Thus in 5 iterations letters from A to E are stored in the array.

13.Ans)

a. (arr + 3)

b. *p

14.Ans) Only (e) and (f) are valid

15.Ans)

x= 32

y= 500

z= 500

16.Ans)

a)n/2

b)i++

c)j--

17.Ans)

```
#include <simpecpp>

main_program{
	int a[20];
	for(int i=0; i<20; i++)
		cin >> a[i];
	int max = a[0];
	for(int i=1; i<20; i++) if(a[i]>max)
		max = a[i];
	cout << max << endl;
}
```

18.Ans) It prints some random number which is the address of variable b

19.Ans) Both 1 and 2

Explanation:- The first two segments signify the same steps in different ways. In the first segment, we divide after finding out the sum. In second segment, we add the weighted data in the loop itself. While in third segment, the variance is incorrect. When we write variance= variance / n - 1; We mean that first division operation takes place, which is followed by the subtraction operation. This makes it incorrect. The correct way to write this is :- variance= variance/(n - 1);

20.Ans) Both func1 and func2

Explanation:- func1: This function directly evaluates the value term by term (addition and multiplication), Eg: p(α) = a0 + a1* α + a2*(α^2) + a3*(α^3)Each term is calculated in inner loop and the terms are summed using the outer loop.

func2: p(α) = a0 + a1* α + a2*(α^2) + a3*(α^3) = (((a3*α + a2) *α + a1) *α + a0).The expression is evaluated in one loop.

21.Ans) (B) ptr1 points to b

ptr1 initially stores address of a and sptr(pointer to pointer) stores address of ptr1. But in the last line, the value in the location(value stored in ptr1) stored by sptr is changed to the value contained in ptr2 ,i.e. address of b

22.Ans)

```
#include<simplecpp>
main_program
{
	int n,i,sum=0;
	cout<<"Enter number of elements you want to insert ";
	cin>>n;
	int arr[n];

	for(i=0;i<n;i++)
	{
		cout<<"Enter element "<<i+1<<": ";
		cin>>arr[i];
		sum+=arr[i];
	}
	cout<<"The sum of Array is : "<<sum<<endl;
	cout<<"The average of Array is :"<<sum/i<<endl;
}
```