Home Work No. 10 (Functions-I)

Codes practice during lectures:

Read comments for the following codes and execute for more understanding. Perform the task written at the end of each practice codes.

////……Code1.cpp……//

```
#include <iostream>

using namespace std;

void sum (void);// 1)prototype/Deceration of non-value returning and parameter-less sum() function

int main()
{
	cout << "\nInside main () function";
	sum();//calling sum () function 1st time
	cout << "\n\nBack to main () function";
	//sum();//calling sum () function 2nd time


}

void sum(void)//2) Defination of function
{
	short val1, val2;//Local variables of sum () function
	int total;
	cout << "\n.....In sum () function..... ";
	cout << "\nEnter Value 1  :   ";
	cin >> val1;
	cout << "\nEnter Value 2  :   ";
	cin >> val2;

	total = val1 + val2;
	cout << "\n Sum of both values is  :  "<< total;

}
```

Task: Write and add non-value returning and parameter-less functions for subtraction, multiplication, division and remainder operations.

////……Code2.cpp……//

```cpp
#include <iostream>

using namespace std;

void sum (int, int );// 1)prototype/Deceration of non-value returning and parameterized
sum() function

int main()
{
	int a1 = 10, a2 = 5;
	cout << "\nInside main () function";
	sum(5,4);//calling sum () function 1st time
	cout << "\n\nBack to main () function";
	sum(a1, a2);//calling sum () function 2nd time


}

void sum(int para1, int para2)//2) Defination of non-value returning and parameterized
sum () function
{
	int total;
	cout << "\n.....In sum () function..... ";

	total = para1 + para2;
	cout << "\n Sum of both values is  :  "<< total;

}
```

Task: Write and add non-value returning and parameterized functions for subtraction, multiplication, division and remainder operations.

////……Code3.cpp……//

```cpp
#include <iostream>

using namespace std;

int sum (int, int );// 1)prototype/Deceration of value returning and parameterized sum() function

int main()
{
	int a1 = 10, a2 = 5;
	int total;
	cout << "\nInside main () function";
	cout<<"\nSum of values is   :"<<sum(5,4);//calling sum () function 1st time
	cout << "\n\nBack to main () function";
	total = sum(a1, a2);//calling sum () function 2nd time
	cout << "\nSum of values is   :" << total;

}
int sum(int para1, int para2)//2) Defination of value returning and parameterized sum () function
{
	return para1 + para2;


}
```

Task: Write and add non-value returning and parameterized functions for subtraction, multiplication, division and remainder operations.

Dry run following codes and explain what will be the output? If you found any error, understand it and then correct the code.

1)

```
int majic(int value)
{
        value += ++value + ++value;
        return value;
}
int main()
{
        int i, j;
        for (i = 0; i <= 5; i++)
        {
                j = majic(i);
                cout << "J =" << j << endl;
        }
        return 0;
}
```

2)

```
int  mystery ( int x , int  n)
{
 int val;
 val =1;
 if (n>0)
  {
    if (n%2 == 1)
    val = val *x;
  }
return val;

}
void main(){
        cout<<"The mysterious value is: "<<mystery(2,3);

}
```

3)

```
float Mystery(int y, int x)
{
	return (y + x + 7.0 / 2);
}

int main()
{
	float i = 9.5;
	int j = 4;
	cout << Mystery(i, j) << endl;
	return 0;
}
```

4)

```
int fun(int x){
	return x % 3 + 1;
}
int main()
{
	int b = 5;
	int y = 2 + fun(3 * b + 1);
	int z = fun(fun(y));
	cout << y << "-" << z;
}
```

5)
What is purpose of following code:

```
int fun(int m, int n)
{
	while (n != m)
	{
		if (n > m)
			n = n - m;
		else if (m > n)
			m = m - n;
	}
	return n;
}
int main()
{

	cout << fun(88, 33) << endl;
	cout << fun(172,140)<<endl;
```

```
        return 0;
        }
```

6)

```
int  mystery(int x, int  n)
{
        int val;
        val = 1;
        if (n >= 0)
        {
                if (n % 3 > 1)
                        val = val * x;
                else
                        val = val * 2;
        }
        return val;

}
void main()
{
cout << "The mysterious value is: " << mystery(10, 3);

}
```

7)

```
        void fun4()
        {
           cout<<"-";
        }
        void fun3(){
           cout<<"+";
           fun4();
           cout<<"+";
        }
        void fun2(){
           cout<<"/";
           fun3();
           cout<<"/";
        }
        void fun1(){
           cout<<"*";
           fun2();
           cout<<"*";
```

```
}
int main(){
   fun1();
   return 0;
}
```

8)

```
int function(int x){
   cout<<"int";
}
int function(float x){
   cout<<"float";
}
int main(){
        double x;
        function(x);
}
```