Home Work No. 11 (Functions-II)

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.

```cpp
////////////////////////////////////code_1.cpp ////////////////////////////
//////////code for Pass by reference parameters/////////////////////


#include <iostream>

using namespace std;

int sum(int&, int&);// 1)prototype/Deceration of value returning and Pass by reference
sum() function

int main()
{
        int a1 = 30, a2 = 15;
        int total;
        cout << "\nInside main () function\n";
        total = sum(a1, a2);//calling sum() function
        cout << "\nSum of values is   :" << total <<endl;

}
int sum(int& para1, int& para2)//2) Defination of value returning and pass by reference
sum () function
{
        cout << "\nInside sum () function\n";

        return para1 + para2;


}
```

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

```cpp
/////////////////////////////////code_2.cpp /////////////////////////
//////////code for default parameters//////////////////////////////


#include <iostream>
#include<iomanip>

using namespace std;

int sum(int =2, int =3);// 1)prototype/Deceration of value returning and parametereized
sum() function with default parameters

int main()
{
        int a1 = 10, a2 = 5;
        int total;
        cout << "\nInside main () function\n";
        total = sum();//calling sum() function for both default parameters
        cout << "\nSum of values is for both default parameters  : " << total << endl;

        total = sum(a1);//calling sum() function for one default parameter
        cout << "\nSum of values is  for one default parameter : " << total << endl;

        total = sum(a1, a2);//calling sum() function
        cout << "\nSum of values is  given arguments : " << total <<endl;

}
int sum(int para1, int para2)//2) Defination of value returning and pass by reference sum
() function
{
        cout << "\nInside sum () function\n";

        return para1 + para2;


}
```

Task: Write functions with defaults parameter for subtraction, multiplication, and 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 counter (int value)
{
 static int count =0;
 count = count +value;;
 return count;
 }
int main()
{
 int i , j;
 for (i=0; i <=5; i++)
   j = counter(i);
   cout<<"J =" <<j<<endl;
 return 0;
}
```

2)

```
void function(int [][3] );

int main()
{
 int a [3][3]= { {1,2,3} , {4,5,6} , {7,8,9} };
 function(a);
cout<< a[2][1];
 return 0;
}

void function ( int b [ ] [3] )
 {
   ++ b;
   b[1][1] =9;
 }
```

3)

```
void e(int );
int main( )
 {
    int a;
    a=3;
    e(a);
    e(a);
    getchar();
 }

void e(static int n)
{
  if(n>0)
      cout<<n<<" ";
      n--;
}
```

4)

```
void find(int a, int& b, int& c);
int main()
{
        int one, two, three;
        one = 5;
        two = 10;
        three = 15;
        find(one, two, three);
        cout << one << ", " << two << ", " << three << endl;
        find(two, one, three);
        cout << one << ", " << two << ", " << three << endl;
        find(three, two, one);
        cout << one << ", " << two << ", " << three << endl;
        find(two, three, one);
        cout << one << ", " << two << ", " << three << endl;
        return 0;
```

```
}
void find(int a, int& b, int& c)
 {
        int temp;
        c = a + 2 * b;
        temp = b;
        b = a;
        a = 2 * temp;
}
```

5)

```
void myFunc(char = '*', int = 10);

int main()
{
   myFunc();
   myFunc('@');
   myFunc ('$', 6);

   return 0;
}

void myFunc(char c, int n)
{
   for(int i = n; i <= 1; i=i-2)
   {
      cout << c;
   }
   cout << endl;
}
```
6)

```
void myFunc(char = '*', int = 10);

int main()
{
   myFunc();
   myFunc('@');
   myFunc ('$', 6);
```

```cpp
    return 0;
}

void myFunc(char c, int n)
{
   for(int i = n; i <= 1; i=i-2)
   {
      cout << c;
   }
   cout << endl;
}
```

7)

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

8)

```cpp
    void e(int n)
    {
            static int x = 3;
            if (n > 0)
                    cout << n << " "<< x<<endl;
            n--;
            x--;
    }
    int main()
    {
            int a;
            a = 10;
```

```
            e(a--);
            e(a--);
            e(a--);


    }

9)
    void RealFun(int a)
    {
            if(a == 1)
                    return;
            cout<<a;
            RealFun(a-1);
            cout<<a;
    }


    int main()
    {
            int a = 4;
            RealFun(a);
     return 0;
    }


10)

    void fun3(int&a)
    {
            a++;
       cout<<a;
    }
    void fun2(int &a){
       fun3(++a);
       cout<<a;
    }
    void fun1(int &a){
       fun2(++a);
       cout<<a;
    }
    int a=5;
    int main(){
       int a = 1;
       fun1(a);
       cout<<a;
       return 0;
    }
```

11)

```cpp
int hello(int a, int &b, int &c) {
        int x;
        b *= 10;
        x = a * b;
        a += 2;
        c++;
        x -= c;
        return x;
}
int main() {
        int a = 10, b = 11, c = 12, result;
        cout << a << " " << b << " " << c << endl;
        result = hello(a, b, c);
        cout << result<<endl;
        return 0;
}
```