Lab 3

Q1. WAP to find the area of a circle, rectangle & a triangle using function overloading.

→

```
#include< iostream>
#include<math>
using namespace std;
double area(int);
int area (int, int);
double area (int, int, int);
int main() {
        int a, b, c, r;
cout << "Enter the length of radius: ";
cin >> r;
cout << " Area of the circle = "<< area(r);
cout << "\n Enter the length of rectangle: ";
cin >> a;
cout << " Enter the breadth of rectangle: ";
 cin >> b;
cout << " Area of the rectangle = "<< area(a, b);
cout << "\n Enter the lengths of the sides of the
          triangle : ";
cin >> a >> b >> c;
 Cout <<" Area of the triangle = "<< area(a, b, c);
}
  double area(int r){
       return 3.14 * r*r;
    }
double area (int a, int b, int c) {
    double s = (a+b+c) / 2;
  }
```

```
int area (int a, int b) {
        return a * b;
}
```

Output

```
Enter the length of radius: 4
Area of the circle: 50.24
Enter the length of rectangle: 5
Area of the rectangle = 15
Enter the lengths of the sides of triangle: 5 12 13
Area of the triangle = 30
```

Q2: WAP to find the volume of a sphere, cylinder & a cuboid using function overloading.

→
```
#include<iostream>
#include<math.h>
using namespace std;
double volume(int);
double volume(int, int);
int volume(int, int, int);
int main() {
    int l, b, h;
    int r;
cout << "Enter the length of radius: ";
cin>> r;
cout << " Volume of the sphere = "<< volume(r);
cout << "\n Enter the radius of the base of the
        cylinder: ";
cin >> r;
cout << "Enter the height of the cylinder: ";
cin >> h;
cout << "Volume of the cylinder = " << volume(r, h);
cout << "\n Enter the length of the cuboid: ";
 cin>> l;
cout << " Enter the breadth of the cuboid: ";
 cin >> b;
cout << " Enter the height of the cuboid: ";
 cin >> h;
cout << " Volume of the cylinder = "<< volume(l, b, h);
}
```

```
double volume(int r) {
    return (4/3.0) * 3.14 * r*r*r;
}
double volume (int r, int h) {
    return 3.14 * r*r* h;
}
int volume (int l, int b, int h) {
        return l * b * h;
}
```

OUTPUT

```
Enter the length of radius: 3
Volume of the sphere = 113.04
Enter the radius of the base of the cylinder: 4
Enter the height of the cylinder : 6
Volume of the cylinder = 301.44
Enter the length of the cuboid: 2
Enter the breadth of the cuboid: 3
Enter the height of the cuboid: 5
Volume of the cylinder = 30
```

Q3 WAP which displays a given character n number of times using a function. When the n value is not provided, it should print the given character 80 times. When both, the character & the n value are not provided, it should print '*' 80 times.

(a) By using function overloading

→
```
# include<iostream>
using namespace std;
void display();
void display(char);
void display(char, int);
int main() {
        char a = 'r';
        int n = 20;
cout << endl;
display();
cout << endl;
display(a);
cout << endl;
display(a, n);
}
void display() {
    for(int i=0; i<10; i++) {
        cout << " * ";
    }
void display(char a) {
    for(int i=0; i<10; i++) {
        cout << a << " ";
    }
}
```

```
void display (char a, int n) {
    for(int i=0; i<n; i++) {
        cout << a << " ";
    }
}
```

OUTPUT

```
* * * * * * * * * *
r r r r r r r r
r r r r r r r r r r r r r r r r r r r r
```

(b) Using default arguments

→
```
#include <iostream>
using namespace std;
void display (char a = '*', int n=10) {
    for (int i=0; i<n; i++) {
        cout << a << " ";
    }
}
int main() {
    display();
    cout << endl;
    display ('s', 9);
    cout << endl;
    display ('s');
}
```

OUTPUT

```
* * * * * * * * * *
s s s s s s s s s
s s s s s s s s s s
```

Q4 WAP to find the square & cube of a number using inline function.

→
```
#include<iostream>
using namespace std;
inline int sq(int a) {
    return a*a;
}
inline int cu(int a) {
    return a*a*a;
}
int main() {
    int n;
cout << "Enter a number : ";
cin >> n;
cout << "The square of the number is : " << sq(n) << endl;
cout << "The cube of the number is: " << cu(n) << endl;
    return 0;
}
```

OUTPUT

```
Enter a number : 5
The square of the number is : 25
The cube of the number is: 125
```

Q5 WAP to increase the value of an argument given to function using inline function.

→

```
#include <iostream>
using namespace std;
inline int incr(int a) {
        return ++a;
    }
int main() {
        int n;
cout << "Enter a number: ";
 cin >> n;
cout << incr(n);
 return 0;
}
```

OUTPUT

```
Enter a number: 4
5
```

Q6 WAP to create a class called 'complex' & implement the following overloading functions 'add' that return a complex number.

(a) complex add (int a, complex s2)
(b) complex add (complex s1, complex s2)

→
```
#include <iostream>
using namespace std;
class complex {
    int r, i;
public:
    void input() {
        cout << "Enter the real part: ";
        cin >> r;
        cout << "Enter the imaginary part: ";
        cin >> i;
    }
complex add(complex n1, complex n2) {
        complex temp;
        temp.r = n1.r + n2.r;
        temp.i = n1.i + n2.i;
        return temp;
    }
void output() {
    cout << "Sum = "<<r << " + " <<i<< "i" <<
        endl;
    }
};
```

```
int main() {
    Complex n1, n2, n3, sum;
    cout << "for first number..\n";
    n1.input();
    cout << "for second number..\n";
    n2.input();
   sum = n3.add(n1, n2);
    sum.output;
   sum = n3.add(2, n2);
    sum.output();
   return 0;
}
```

OUTPUT

```
for first number..
Enter the real part: 4
Enter the imaginary part: 5
for second number..
Enter the real part: 2
Enter the imaginary part: 3
Sum = 6 + 8i
Sum = 4 + 3i
```

Q7 WAP to find the sum of 3 numbers using one function & default arguments.

→
```
#include<iostream>
using namespace std;
int sum(int a, int b = 10, int c = 20){
    return a+b+c;
}
int main() {
    cout<<"Sum = "<<sum(10,10,10)<<endl;
    cout<<"Sum = "<<sum(10,5)<<endl;
    cout<<"Sum = "<<sum(10)<<endl;
    return 0;
}
```

OUTPUT

```
Sum = 30
Sum = 35
Sum = 40
```

Q8. WAP to demonstrate the concept of call-by-value, call-by-reference & call-by-address by swapping two numbers.

→
```
#include<iostream>
using namespace std;
void swapvalue(int a, int b);
void swapref(int &a, int &b);
void swapad(int *a, int *b);
int main() {
    int x, y;
    cout << "Enter the value of x: ";
    cin >> x;
    cout << "Enter the value of y: ";
    cin << y;
    swapvalue(x,y);
    swapref(x,y);
    swapad(&x, &y);
}
void swapvalue(int a, int b) {
        int c;
        c= a;
        a = b;
        b = c;
cout << "After swapping, x: "<<a<<" & y = "<<
        b << endl;
}
```

```
void swapref (int &a, int &b) {
        int c;
        c = a;
        a = b;
        b = c;
cout << "After swapping, x= " << a << " &y = " <<
      b << endl;
  }
void swapad(int *a, int *b) {
        int c;
        c = *a;
        *a = *b;
        *b = c;
cout << "After swapping, x= " << *a << " &y = " <<
     *b << endl;
  }
```

OUTPUT

```
Enter the value of x: 1
Enter the value of y: 2
After swapping, x=2 &y=1
After swapping, x=2 & y=1
After swapping, x=1 & y=2
```

Name Sanjam Sah

Roll 2105318

Sec. CSE 17