Object Oriented Programming in C++

Nepal College of Information Technology

Course Instructor: Er. Rabina Chaudhary

Introduction of Class:

- Class is user defined data type that is used to specify data representation and methods for manipulating the data in one package.
- The data and functions within a class are called members of the class.
- When you define a class, you define blueprint for a data type.
- Class defines what an object of class will contain and what operations can be performed in it.
- The data members and member functions can be grouped in private, public or protected section.

Class:

- Syntax for defining a class;

```
class class_name
{
        access specifier:
                variable_declaration;
                function_declaration;
         access specifier:
                variable_declaration;
                function_declaration;
};
```

- Example:

```
class Student
{
        private:
                char name[20];
                int roll;
                int marks;
        public:
                void getDetails();
                void display();
};
```

Data hiding in C++

- Data hiding is technique used in object oriented programming to hide object details i.e. data members to limit access to data and prevent them from unwanted manipulation
- There are three access specifiers:
  i. private
  ii. public
  iii. protected

i. Private access specifier:

- If the class members are declared private, then they can be accesses by member functions of that class only.
- Data members are made private to prevent direct access from outside the class.

ii. Public access specifier:

- If the class members are declared public then they can be accessed from anywhere in the program.
- Member functions are usually public which is used to manipulate the data present in the class.

iii. Protected access specifier:

It is similar to that of private access modifiers, the difference is that the class member declared as Protected are inaccessible outside the class but they can be accessed by any derived class of that class.

Creating the object:

Syntax:

```
class_name object_name;

Student s1; //this statement creates a variable s1 of type Student
            //the variable s1 is object of class Student

Student s1,s2,s3,s4; // defining multiple objects of class Student
```

Creating the object:

```
class Student
{
    private:
        char name[20];
        int roll;
        int marks;
    public:
        void getDetails();
        void display();
    }s1,s2;
```

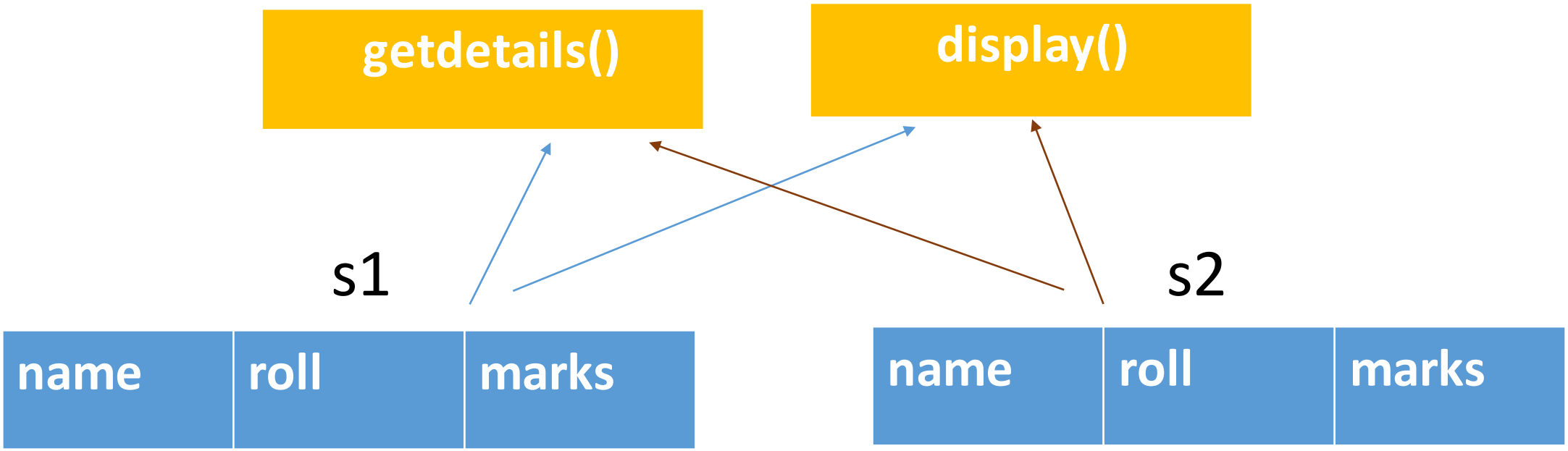
Creating the object:

```
class Student
{
    private:
            char name[20];
            int roll;
            int marks;
    public:
            void getDetails();
            void display();
};
```

```
void main()
    {
            Student s1,s2;
    ...
    }
```

Creating the object:

Student s1,s2;



Accessing class members:

- The private members of a class can be accessed only through members of same class
- The public members of a class can be accessed from outside the class using object name and dot operator

Syntax for accessing public data member:

object_name.data_member;

Syntax for accessing public member functions:

object_name.function_name(arguments);

Accessing class members: Example

```
class Student
{
    private:
        char name[20];
        int roll;
        int marks;
    public:
        void getDetails();
        void display();
};
```

```
void main()
{
        Student s1;
        s1.getDetails();
        s1.display();

        s1.roll=14578; //invalid as roll is private member
```

Accessing class members: Example

```
class Student
{
    private:
        char name[20];
        int marks;
    public:
        int roll;
        void getDetails();
        void display();
};
```

```
void main()
{
    Student s1;
    s1.getDetails();
    s1.display();
```

s1.roll=14578; valid as roll is public member

Defining member function:

1. Inside the class body
2. Outside the class body

1. Defining member function inside class body

If we define member function inside class definition, it is inline function

Defining member function: inside class body

```
class Student
{
    private:
            char name[20];
            int roll;
            int marks;
    public:
            void getDetails()
            {
            cout<<“Enter details”;
            cin>>name>>roll>>marks;
            }
            void display()
            {
            cout<<“The name is “<<name<<endl;
            cout<<“Roll is “<<roll<<endl;
            cout<<“Marks is “<<marks<<endl;
            }
};
```

Defining member function: outside class body

- The function prototype is defined within the class body and its detail definition is written outside class body
- If we define function outside class body, we use scope resolution **::** operator

Syntax:

```
return_type class_name :: function_name(arguments)
{
        function body
}
```

Defining member function: outside class body

```
class Student
{
    private:
            char name[20];
            int roll;
            int marks;
    public:
            void getDetails();
            void display();
    };
```

```
void Student :: getDetails()
 {
  cout<<“Enter details”;
  cin>>name>>roll>>marks;
 }
 void Student :: display()
 {
cout<<“The name is “<<name<<endl;
cout<<“Roll is “<<roll<<endl;
cout<<“Marks is “<<marks<<endl;
}
```

Example Program:

// class and object example program
//member function defined inside class body

```
#include<conio.h>
#include<iostream.h>

class Student
{
        private:
                char name[20];
                int roll;
                int marks;
        public:
        void getDetails()
        {
                cout<<"Enter name of student : ";
                cin>>name;
                cout<<"Enter roll number : ";
                 cin>>roll;
                 cout<<"Enter marks obtained :";
                  cin>>marks;
        }
void display()
                {
                cout<<endl<<endl<<"The  Details are :"<<endl;
                cout<<"The  name is "<<name<<endl;
                cout<<"Roll is "<<roll<<endl;
                cout<<"Marks  is "<<marks<<endl;
                }
};
```

Program continued…

```
void main()
{
  Student S1,S2;
    S1.getDetails();
    cout<<endl;
    S2.getDetails();
    cout<<endl;
    S1.display();
    cout<<endl;
    S2.display();
    getch();
}
```

Note: Compile and run this program.

Example Program:

```
// class and object example program
//member  function defined outside class body
#include<conio.h>
#include<iostream.h>

class Student
{
        private:
                char name[20];
                int roll;
                int marks;
        public:
                void getDetails();      //member function declaration inside class.
                void display();

};
```

Program continued…

```
void Student:: getDetails()              //member function definition outside class
{
                cout<<endl<<"Enter name of student : ";
                cin>>name;
                cout<<"Enter roll number : ";
                 cin>>roll;
                 cout<<"Enter marks obtained :";
                cin>>marks;
}
void Student :: display()                //member function definition outside class
{
                cout<<endl<<endl<<"The Details are :"<<endl;
                cout<<"The name is "<<name<<endl;
                cout<<"Roll is "<<roll<<endl;
                cout<<"Marks is "<<marks<<endl;
}
```

Program continued…

```
void main()
{
  Student S1,S2;     //object creation
   S1.getDetails();  //public member function call
   S2.getDetails();
   S1.display();
   S2.display();
   getch();
}
```

Note:Compile and run the program.

Practice:

Q1. Design a class called Person that contains appropriate members for storing name, age, gender, telephone number. Write member functions that can read and display these data.

Q2. Write a program to represent a Circle that has member functions to perform following tasks.

- Calculate area of circle
- Calculate perimeter of the circle

Q3. Create a class Point that represents a three dimensional coordinate system. Each object of Point should have coordinates x,y,z and methods to assign coordinates to the objects. Add a method to calculate the distance from origin and to the point (x,y,z). Define member functions outside the class body.

Q1. Design a class called Person that contains appropriate members for storing name, age, gender, telephone number. Write member functions that can read and display these data. Input details of 3 Person and display their details.

Nesting of member functions:

- A member function of a class can be called only by an object of that class using dot operator
- However, a member function can be called by using its name inside another member function of same class
- This is known as nesting of member function

Nesting of member function : Example

```
class Student
{
        private:
                char name[20];
                int roll;
                int marks;
        public:
                void studentDetails();
                void display();
};
```

```
void Student:: studentDetails()
{
..
 display();
}


 void main()
{
        Student S;
        S.studentDetails();
        …
}
```

Array of Objects:

Syntax for declaring array of objects:

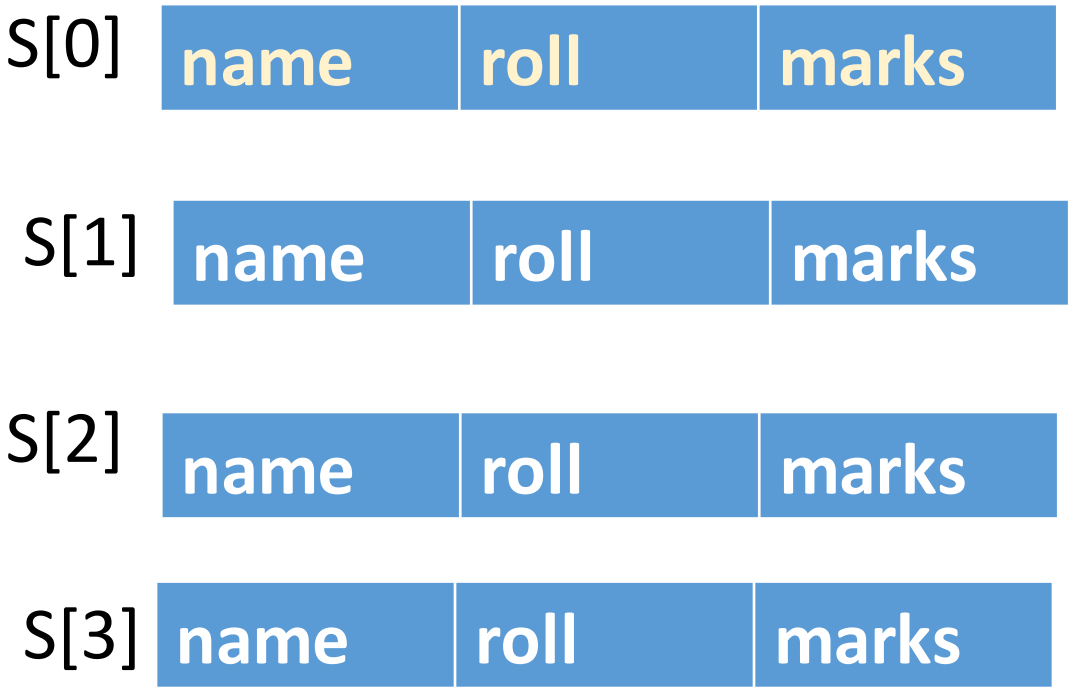
```
class_name  object_name[size];
```

Example:

```
Student S[50];
```

Array of Objects : Example Program

Student S[4];



Example Program:

```
#include<conio.h>
#include<iostream.h>

class Student
{
        private:
                char name[20];
                int roll;
                int marks;

        public:
                void getDetails(int);
                void display();

};
```

Program continued…

```
void Student:: getDetails(int x)
{
                cout<<endl<<"Enter details of student "<<x<<" : "<<endl;
                cout<<"Name          : ";
                 cin>>name;
                cout<<"Roll number     : ";
                cin>>roll;
                cout<<"Marks obtained :";
                cin>>marks;
                cout<<endl<<endl;
}
void Student :: display()
{
                cout<<"The name is "<<name<<endl;
                cout<<"Roll is "<<roll<<endl;
                cout<<"Marks is "<<marks<<endl<<endl;
}
```

Program Continued…

```
void main()
{
		Student S[50];
		int i,n;
		cout<<"Enter number of Students : ";
		cin>>n;
		cout<<endl<<endl<<"Enter details of students"<<endl;
		for(i=0;i<n;i++)
		{
			S[i].getDetails(i+1);
		}
		cout<<endl<<endl<<"Details of student are :"<<endl;
		for(i=0;i<n;i++)
		{
			S[i].display();
		}
  getch();
}
```

Practice:

Q1. Write a program to define a class named Employee with its data members name, salary, id number. Read records of n number of employees and display the records.