

# UNIVERSITY OF ENGINEERING AND TECHNOLOGY (NAROWAL CAMPUS)



## Object-Oriented Programming Lab Manual

Created by: Muhammad Abdullah

Registration Number: 2022-CS-525

Topics: Vectors, Pass Structure to Function,  
The Array of Structures, Structure in Structure,  
And Pointer with Structures

# Lab Manual

## (Object-Oriented Programming Lab)

---

- **Vectors:**

A vector is just simply a smart array that can manage storage allocation, expanding and contracting the size of the vector as you insert or erase data. We can use vectors much like arrays, accessing elements with the subscript operator. The header file used for vectors is <vector> which allows us to use vectors and also provides the built-in function for the operations of vectors. The back of the vector is where the element with the highest index number is. The front of the vector is where the element with the index 0. The syntax of the vector is as follows:

```
vector <dataType> Identifier;
```

The above statement declares a vector. We can also provide the size of the vector at the time of declaration. The statement is given as follows:

```
vector <int> List (25);
```

The above statement declares a vector of integer datatype and its size is 25.

### **Functions of <vector> header file:**

The vector header file provides the functions for the operations of vectors. The details of some functions which is frequently used are as follows:

#### **1. push\_back() function:**

The push\_back function inserts the value of its argument at the back of the vector. The syntax is as follows:

```
List.push_back(525);
```

#### **2. size() function:**

The size function returns the number of elements currently present in the vector. The syntax is as follows:

```
List.size();
```

#### **3. max\_size() function:**

The max\_size function returns the maximum size to which a vector can expand. This number depends on the type of data being stored in the container the bigger the elements, the fewer of them you can store, the type of container, and the operating system. The syntax is as follows:

```
List.max_size();
```

#### **4. swap() function:**

The swap function exchanges all data in one vector with all the data in another keeping the elements in the same order. The syntax is as follows:

```
List.swap(List2);
```

#### **5. back() function:**

The back function returns the value of the last element in the vector. The syntax is as follows:

```
List.back();
```

#### **6. pop\_back() function:**

The pop\_back function removes the last element in the vector. The syntax is as follows:

```
List.pop_back();
```

#### **7. at() function:**

The at function returns the value which is at the parameter value index. The syntax is as follows:

```
List.at(17);
```

#### **8. front() function:**

The back function returns the value of the first element in the vector. The syntax is as follows:

```
List.front();
```

#### **9. clear() function:**

The clear function deletes all the elements from the vector. The syntax is as follows:

```
List.clear()
```

#### **10. empty() function:**

The empty method returns a Boolean value and tells whether a vector is empty or not. The syntax is as follows:

```
List.empty();
```

#### **11. resize() function:**

The resize function increases the size of the vector and gives the new size of the parameter value. The syntax is as follows:

```
List.resize(50);
```

- **Programs:**

Task 1: Pass a structure to a function both by value and by reference

Task 2: Create an array of Structures and print each member's value

Task 3: Create a nested Structure and access its values

Task 4: Access the member value of a Structure using Pointer

Task 5: Apply operation on a vector

**Task 1-A Program:**

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4  struct student{
5      string name;
6      string fname;
7      int age;
8      int rollno;
9  };
10 void print(student);
11 int main(){
12     student student;
13     cout<<"Enter Your Name: ";
14     getline(cin, student.name);
15     cout<<"Enter Your Father Name: ";
16     getline(cin, student.fname);
17     cout<<"Enter Your Age: ";
18     cin>>student.age;
19     cout<<"Enter Your Roll Number: ";
20     cin>>student.rollno;
21     print(student);
22     return 0;
23 }
24 void print(student candidate){
25     cout<<"Student Name: "<<candidate.name<<endl;
26     cout<<"Student Father Name: "<<candidate.fname<<endl;
27     cout<<"Student Age: "<<candidate.age<<endl;
28     cout<<"Student Roll Number: "<<candidate.rollno<<endl;
29 }
```

**Task 1-A Output:**

```
Enter Your Name: Muhammad Abdullah
Enter Your Father Name: Zahid Mehmood
Enter Your Age: 18
Enter Your Roll Number: 525
Student Name: Muhammad Abdullah
Student Father Name: Zahid Mehmood
Student Age: 18
Student Roll Number: 525

D:\UET Narowal\2nd Semester\Object Oriented Programming\Lab\OOP Lab\x64\Debug\OOP Lab.exe (process 13348) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

### Task 1-B Program:

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4  struct student{
5      string name;
6      string fname;
7      int age;
8      int rollno;
9  };
10 void print(student);
11 void change(student &);
12 int main(){
13     student student = {"Abdullah", "Zahid" , 00, 00};
14     print(student);
15     change(student);
16     print(student);
17     return 0;
18 }
19 void change(student &candidate){
20     cout<<" --: Change Information :--"<<endl;
21     cout<<"Enter New Name: ";
22     getline(cin,candidate.name);
23     cout<<"Enter New Father Name: ";
24     getline(cin, candidate.fname);
25     cout<<"Enter New Age: ";
26     cin>>candidate.age;
27     cout<<"Enter New Roll Number: ";
28     cin>>candidate.rollno;
29 }
30 void print(student candidate){
31     cout<<"Student Name: "<<candidate.name<<endl;
32     cout<<"Student Father Name: "<<candidate.fname<<endl;
33     cout<<"Student Age: "<<candidate.age<<endl;
34     cout<<"Student Roll Number: "<<candidate.rollno<<endl;
35 }
```

### Task 1-B Output:

```
Student Name: Abdullah
Student Father Name: Zahid
Student Age: 0
Student Roll Number: 0
--: Change Information :--
Enter New Name: Reshial Kareem
Enter New Father Name: Abdul Kareem
Enter New Age: 19
Enter New Roll Number: 500
Student Name: Reshial Kareem
Student Father Name: Abdul Kareem
Student Age: 19
Student Roll Number: 500

D:\UET Narowal\2nd Semester\Object Oriented Programming\Lab\OOP Lab\Debug\OOP Lab.exe (process 17196) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

## Task 2 Program:

```
1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4  struct soilder
5  {
6      string name;
7      string rank;
8      string country;
9  };
10 void print(soilder[], int);
11 int main()
12 {
13     int n = 5;
14     soilder team[5] = {
15         {"Muhammad Abdullah", "Lt. Col.", "Pakistan"},
16         {"Tahir Masood Bhutta", "Maj. Gen.", "Pakistan"},
17         {"Tariq Mehmmod", "Brig. Gen.", "Pakistan"},
18         {"Napoleon Boneparte", "Gen.", "France"},
19         {"Ali Sarwar", "Maj.", "Pakistan"}};
20     print(team, n);
21     return 0;
22 }
23 void print(soilder array[], int n)
24 {
25     for (int i = 0; i < n; i++)
26     {
27         cout<<"--:The Details of Number "<<i+1<<" Team Member:--"<<endl;
28         cout<<"Name: "<<array[i].name<<endl;
29         cout<<"Rank: "<<array[i].rank<<endl;
30         cout<<"Country "<<array[i].country<<endl<<endl;
31     }
32 }
33
```

## Task 2 Output:

```
--:The Details of Number 1 Team Member:--
Name: Muhammad Abdullah
Rank: Lt. Col.
Country Pakistan

--:The Details of Number 2 Team Member:--
Name: Tahir Masood Bhutta
Rank: Maj. Gen.
Country Pakistan

--:The Details of Number 3 Team Member:--
Name: Tariq Mehmmod
Rank: Brig. Gen.
Country Pakistan

--:The Details of Number 4 Team Member:--
Name: Napoleon Boneparte
Rank: Gen.
Country France

--:The Details of Number 5 Team Member:--
Name: Ali Sarwar
Rank: Maj.
Country Pakistan

D:\UET Narowal\2nd Semester\Object Oriented Programming\Lab\OOP Lab\Debug\OOP Lab.exe (process 11580) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

### Task 3 Program:

```
1  #include <iostream>
2  using namespace std;
3  struct basicinfo{
4      string name;
5      string fname;
6      string gender;
7      int age;
8  };
9  struct academicinfo{
10     string dept;
11     int rollno;
12     int session;
13     double cgpa;
14 };
15 struct information{
16     basicinfo basic;
17     academicinfo academic;
18 };
19 void print(information);
20 int main(){
21     information Abdullah = {{ "Muhammad Abdullah" , "Zahid Mehmood" , "Male" , 18},{ "Computer Science",525,2022,3.29}};
22     information AbuBakar = {{ "Muhammad Abu Bakar" , "Zahid Mehmood" , "Male" , 14},{ "Software Engineering" , 00, 2026, 00.0}};
23     print(Abdullah);
24     print(AbuBakar);
25     return 0;
26 }
27 void print(information info){
28     cout<<"Name: "<<info.basic.name<<endl;
29     cout<<"Father Name: "<<info.basic.fname<<endl;
30     cout<<"Gender: "<<info.basic.gender<<endl;
31     cout<<"Age: "<<info.basic.age<<endl;
32     cout<<"Department: "<<info.academic.dept<<endl;
33     cout<<"Roll No.: "<<info.academic.rollno<<endl;
34     cout<<"Session: "<<info.academic.session<<endl;
35     cout<<"CGPA: "<<info.academic.cgpa<<endl;
36 }
```

### Task 3 Output:

```
Name: Muhammad Abdullah
Father Name: Zahid Mehmood
Gender: Male
Age: 18
Department: Computer Science
Roll No.: 525
Session: 2022
CGPA: 3.29
Name: Muhammad Abu Bakar
Father Name: Zahid Mehmood
Gender: Male
Age: 14
Department: Software Engineering
Roll No.: 0
Session: 2026
CGPA: 0

D:\UET Narowal\2nd Semester\Object Oriented Programming\Lab\OOP Lab\Debug\OOP Lab.exe (process 5900) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

### Task 4 Program:

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4  struct car{
5      string brand;
6      string model;
7      int hp;
8      int torque;
9      string price;
10 };
11 void print(car*);
12 void change(car*);
13 int main(){
14     car Mycar = {"Ford", "Mustang", 315, 350, "$100 000"};
15     print(&Mycar);
16     change(&Mycar);
17     print(&Mycar);
18     return 0;
19 }
20 void print(car *cars){
21     cout<<"Brand: "<<(cars->brand)<<endl;
22     cout<<"Model: "<<(cars->model)<<endl;
23     cout<<"Horse Power: "<<(cars->hp)<<endl;
24     cout<<"Torque: "<<(cars->torque)<<endl;
25     cout<<"Price: "<<(cars->price)<<endl;
26 }
27 void change(car *car){
28     cout<<"Enter New Car Brand Name: ";
29     getline(cin, car->brand);
30     cout<<"Enter New Car Model Name: ";
31     getline(cin, car->model);
32     cout<<"Enter New Car Price: ";
33     getline(cin, car->price);
34     cout<<"Enter New Horse Power: ";
35     cin>>car->hp;
36     cout<<"Enter New Torque: ";
37     cin>>car->torque;
38 }
```

### Task 4 Output:

```
Brand: Ford
Model: Mustang
Horse Power: 315
Torque: 350
Price: $100 000
Enter New Car Brand Name: Ferrari
Enter New Car Model Name: 425
Enter New Car Price: $900 000
Enter New Horse Power: 463
Enter New Torque: 425
Brand: Ferrari
Model: 425
Horse Power: 463
Torque: 425
Price: $900 000

D:\UET Narowal\2nd Semester\Object Oriented Programming\Lab\OOP Lab\x64\Debug\OOP Lab.exe (process 7512) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```



## Task 5 Program:

```
#include <iostream>
#include <vector>
using namespace std;
int main(){
    vector<int> array(5) ;
    array = {130,425,463,901,947};
    cout<<" --: The Vector Before Operations :--"<<endl;
    for(int i=0;i<array.size();i++){
        cout<<array.at(i)<<" ";
    }

    cout<<endl<<"The first value of vector before operations is: "<<array.front()<<endl;
    cout<<"The last value of vector before operations is: "<<array.back()<<endl;
    cout<<"The size of vector before operations is: "<<array.size()<<endl;
    array.pop_back();
    array.pop_back();
    array.push_back(100);
    array.push_back(200);
    array.push_back(300);
    array.resize(25);
    cout<<" --: The Vector After Operations :--"<<endl;
    for(int i=0;i<array.size();i++){
        cout<<array.at(i)<<" ";
    }

    cout<<endl<<"The first value of vector after operations is: "<<array.front()<<endl;
    cout<<"The last value of vector after operations is: "<<array.back()<<endl;
    cout<<"The size of vector after operations is: "<<array.size()<<endl;
    return 0;
}
```

## Task 5 Output:

```
--: The Vector Before Operations :--
130 425 463 901 947
The first value of vector before operations is: 130
The last value of vector before operations is: 947
The size of vector before operations is: 5
--: The Vector After Operations :--
130 425 463 100 200 300 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
The first value of vector after operations is: 130
The last value of vector after operations is: 0
The size of vector after operations is: 25

D:\UET Narowal\2nd Semester\Object Oriented Programming\Lab\OOP Lab\x64\Debug\OOP Lab.exe (process 6816) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```