Practical Lecture 4: Concepts & Basics of C++ Programming

LOVELY PROFESSIONAL UNIVERSITY

PUNJAB (INDIA)

Quick Recap

Let’s take a quick recap of previous lecture –

A)

B)

C)

D)

E)

F)

Today’s Agenda

Today we are going to cover -

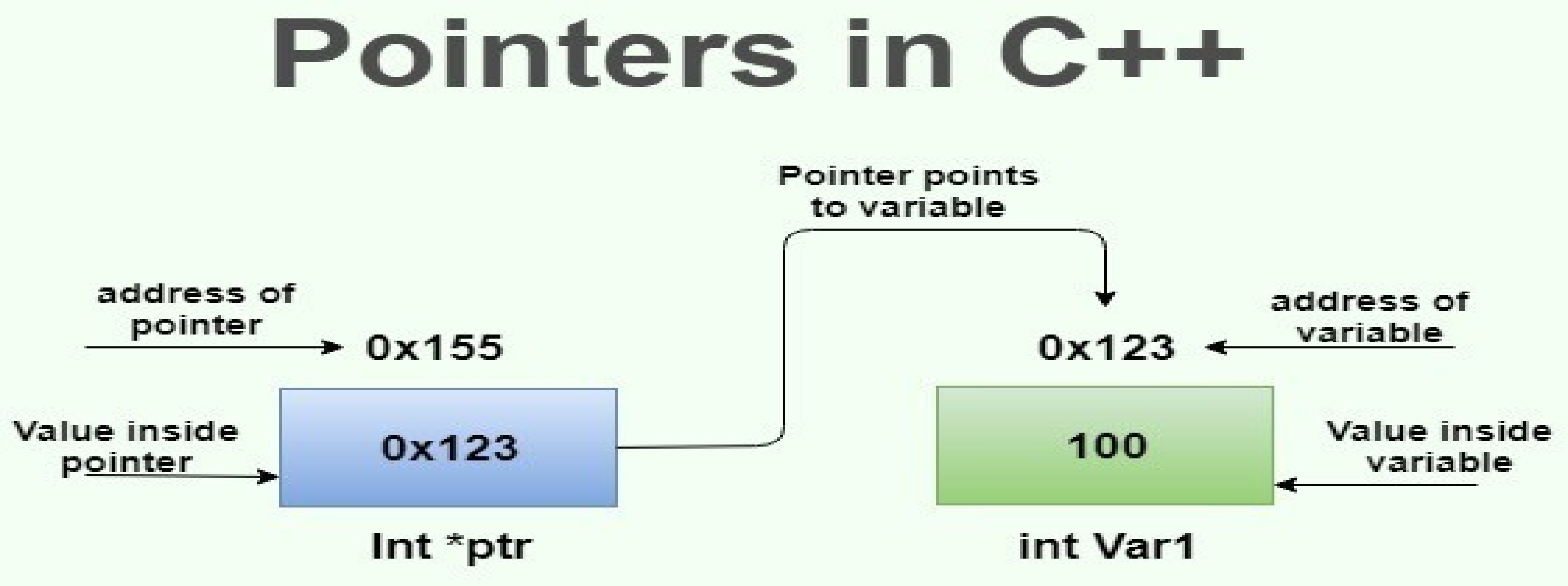
- Pointers
- Structure
- Enum
- Union
- Practice Questions
- Quick Revision

MCQ Questions

Let’s Get Started-

Pointers

Pointers :-The pointer in C++ language is a variable, it is also known as locator or indicator that points to an address of a value.



Pointers

Advantages of Pointers:-

1) Pointer reduces the code and improves the performance, it is used to retrieving strings, trees etc. and used with arrays, structures and functions.

2) We can return multiple values from function using pointer.

3) It makes you able to access any memory location in the computer's memory.

Pointers

```
#include <iostream>
using namespace std;

int main()
{
int number=30;
int ∗  p;
p=&number;//stores the address of number variable
cout<<"Address of number variable is:"<<&number<<endl;
cout<<"Address of p variable is:"<<p<<endl;
cout<<"Value of p variable is:"<<*p<<endl;
   return 0;
}
```

Output

```
Address of number variable is:0x7ffccc8724c4
Address of p variable is:0x7ffccc8724c4
Value of p variable is:30
```

Practice Questions

1.Write a program to swap two number without using third variable using pointers.

2.Write a program in C to store n elements in an array and print the elements using pointer.

Test Data :

```
Input the number of elements to store in the array :5
Input 5 number of elements in the array :
element - 0 : 5
element - 1 : 7
element - 2 : 2
element - 3 : 9
element - 4 : 8
```

Practice Questions

Expected Output :

```
3.The elements you entered are :
element - 0 : 5
element - 1 : 7
element - 2 : 2
element - 3 : 9
element - 4 : 8
```

4.Write a program in C to find the factorial of a given number using pointers.

Test Data :

Input a number : 5

Expected Output : The Factorial of 5 is : 120

Structure

We often come around situations where we need to store a group of data whether of similar data types or non-similar data types. We have seen Arrays in C++ which are used to store set of data of similar data types at contiguous memory locations.

Unlike Arrays, Structures in C++ are user defined data types which are used to store group of items of non-similar data types.

A structure is a user-defined data type in C/C++. A structure creates a data type that can be used to group items of possibly different types into a single type.

Structure Example

```
#include <iostream>
#include<string>
using namespace std;

struct Student{
   string name;
   int stuRollNo;
   int stuAge;
};
```

Structure Example

```
int main(){
  Student s;
  cout<<"Enter Student Name: ";
  cin>>s.name;
  cout<<"ENter Student Roll No: ";
  cin>>s.stuRollNo;
  cout<<"Enter Student Age: ";
  cin>>s.stuAge;
  cout<<"Student Record:"<<endl;
  cout<<"Name: "<<s.name<<endl;
  cout<<"Roll No: "<<s.stuRollNo<<endl;
  cout<<"Age: "<<s.stuAge;
  return 0;
}
```

Structure Practice Question's

A.)Write a structure to store the roll no., name, age (between 11 to 14) and address of students (more than 10). Store the information of the students.

1 - Write a function to print the names of all the students having age 14.

2 - Write another function to print the names of all the students having even roll no.

3 - Write another function to display the details of the student whose roll no is given (i.e. roll no. entered by the user).

B.)Write a structure to store the name, account number and balance of customers (more than10) and store their information.

1 - Write a function to print the names of all the customers having balance less than $200.

2 - Write a function to add $100 in the balance of all the customers having more than $1000 in their balance and then print the incremented value of their balance.

Enum

Enumerated type (enumeration) is a user-defined data type which can be assigned some limited values. These values are defined by the programmer at the time of declaring the enumerated type.
When we assign a float value in a character value then compiler generates an error in the same way if we try to assign any other value to the enumerated data types the compiler generates an error. Enumerator types of values are also known as enumerators. It is also assigned by zero the same as the array. It can also be used with switch statements.

Enum

```
#include <iostream>
using namespace std;

enum direction {East, West, North, South};

int main(){

   direction dir;
   dir = South;
   cout<<dir;
   return 0;
}
```

Enum

```
 #include <bits/stdc++.h>
using namespace std;

enum  year {Jan,Feb,Mar,Apr,May,Jun,Jul,Aug,Sep,Oct,Nov,Dec};

int main()
{
        int i;

        // Traversing the year enum
        for (i = Jan; i <= Dec; i++)
                cout << i << " ";

        return 0;
}
```

What will be the output of the following program

```
#include <iostream>
  using namespace std;
  enum  cat
  {
     temp = 7
  };
  int main()
  {
     int age = 14;
     age /= temp;
     cout << "If you were cat, you would be " << age << endl;
     return 0;
  }
```

What will be the output of the following program

a) If you were cat, you would be 5

b) If you were cat, you would be 2

c) If you were cat, you would be 7

d) If you were cat, you would be 9

Solution

a) If you were cat, you would be 5

b) If you were cat, you would be 2

c) If you were cat, you would be 7

d) If you were cat, you would be 9

Answer: b
Explanation: The age will be divided by using compound assignment operator and so it will return the age of the cat according to your age.

What will be the output of the following program.

```
#include <iostream>
  using namespace std;
  enum test
  {
     A = 32, B, C
  };
  int main()
  {
     cout << A << B<< C;
     return 0;
  }
```

What will be the output of the following question.

a) 323334

b) 323232

c) 323130

d) 323134

Solution

a) 323334

b) 323232

c) 323130

d) 323134

Answer: a
Explanation: If we not assigned any value to enum variable means, then the next number to initialized number will be allocated to the variable.

UNION

```
#include <iostream>

using namespace std;

struct student1 {        // defining a struct
        int roll_no;
        char name[40];
        int phone_number;
};

union student2 {         // defining a union
        int roll_no;
        char name[40];
        int phone_number;
};
```

UNION

```
int main()
{
        struct student1 s1;
        union student2 u1;
        cout << "size of structure : " << sizeof(s1) << endl;
        cout << "size of union : " << sizeof(u1) << endl;
        return 0;
}
```

OutPut:-

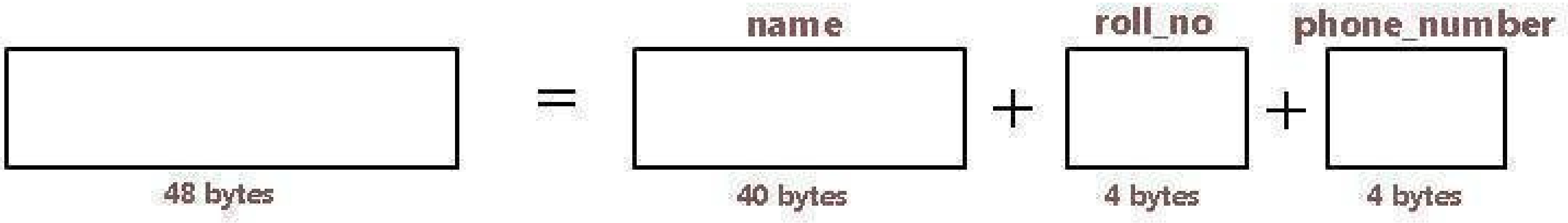
```
size of structure : 48
size of union : 40
```

UNION

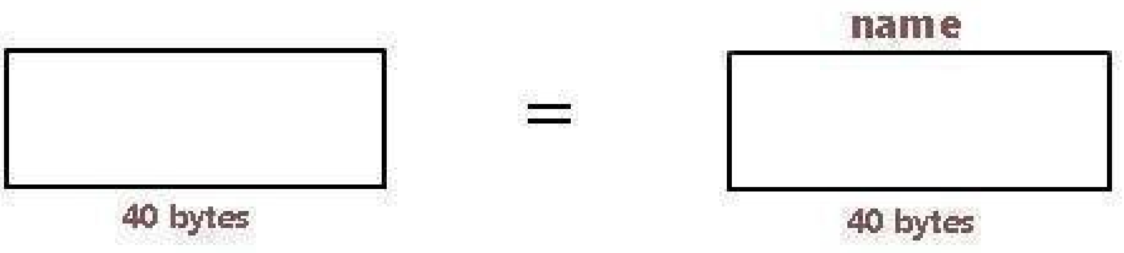
Talking about the above example, the amount of memory required to store a structure is the sum of the memory sizes of all its members. In the above example, the memory sizes of the variables roll_no and phone_number will be 4 bytes each (since both are of type integer) and the memory size of the character array name[40] will be 40 bytes (since the array occupies the memory of 40 characters and the size of char is 1). Thus the memory occupied by the structure will be 4+40+4 = 48 bytes.

Now coming to the union, the memory size of a union is equal to the size of its member occupying the maximum space in the memory. The size of roll_no and phone_number is 4 bytes each and that of name[40] is 40 bytes. So, the union will occupy a memory space of 40 bytes.

UNION



Memory allocated to structure



Memory allocated to union

UNION

Talking about the above example, the amount of memory required to store a structure is the sum of the memory sizes of all its members. In the above example, the memory sizes of the variables roll_no and phone_number will be 4 bytes each (since both are of type integer) and the memory size of the character array name[40] will be 40 bytes (since the array occupies the memory of 40 characters and the size of char is 1). Thus the memory occupied by the structure will be 4+40+4 = 48 bytes.

Now coming to the union, the memory size of a union is equal to the size of its member occupying the maximum space in the memory. The size of roll_no and phone_number is 4 bytes each and that of name[40] is 40 bytes. So, the union will occupy a memory space of 40 bytes.

UNION

We can access only one member of union at a time because we have only one location in memory for it, so only one of the member can be used at a time. All the other members will contain the garbage value (i.e. will get corrupted). This is not the case with structures where we can access all the member's variables at the same time because each occupies a different memory space.

Revision

1. What is pointer's?

2. Explain difference b/w Union and structure .

3. What is enum explain in details.

Assignment for you

1. Write a function which will take pointer and display the number on screen. Take number from user and print it on screen using that function.

2. Declare a structure to represent a complex number (a number having a real part and imaginary part). Write C++ functions to add, subtract, multiply and divide two complex numbers.

QNA Time

Any Questions ??

Any Questions??

Thank You!

See you guys in next class.