National University of Computer and Emerging Sciences



Lab Manual 01

Object Oriented Programming

|  |  |
| --- | --- |
| Course Instructor | Mr. Bismillah Jan |
| Lab Instructor (s) | Mr. Saif Ali  Mr. Dilawar Shabbir |
| Section | 2E |
| Semester | Spring 2021 |

Department of Computer Science

FAST-NU, Lahore, Pakistan

## Objectives

After performing this lab, students shall be able to:

* Have an improved understanding of pointers.
* Access and modify pointers in functions.
* How pointers and array can be related
* Debugging

**TASK 1:**

See the following piece of code and write its output by debugging the code

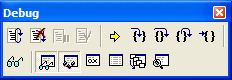
|  |
| --- |
| int myFunction ()  {  int numbers[5];  int \* p;  p = numbers;  \*p = 10;  p++;  \*p = 20;  p = &numbers[2];  \*p = 30;  p = numbers + 3;  \*p = 40;  p = numbers;  \*(p+4) = 50;  for (int n=0; n<5; n++)  cout << numbers[n] << ", ";  return 0;  }  Void main()  {  myFunction();  } |

Write the address of array named ‘numbers’

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 |
|  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sr.No | code | Value of p | Address of p | Value of array ‘numbers’ | | | | |
|  | [0] | [1] | [2] | [3] | [4] |
| 1 | int numbers[5]; |  |  |  |  |  |  |  |
| 2 | int \* p=numbers; |  |  |  |  |  |  |  |
| 3 | \*p = 10; |  |  |  |  |  |  |  |
| 4 | p++; |  |  |  |  |  |  |  |
| 5 | \*p = 20; |  |  |  |  |  |  |  |
| 6 | p = &numbers[2]; |  |  |  |  |  |  |  |
| 7 | \*p = 30; |  |  |  |  |  |  |  |
| 8 | p = numbers + 3; |  |  |  |  |  |  |  |
| 9 | \*p = 40; |  |  |  |  |  |  |  |
| 10 | p = numbers; |  |  |  |  |  |  |  |
| 11 | \*(p+4) = 50; |  |  |  |  |  |  |  |

**Help:** Debugging commands:

** **

|  |  |  |  |
| --- | --- | --- | --- |
| Short cut key | Icon | Menu | Explanation |
| F-9 |  |  | Insert/Remove breakpoint |
| F-5 |  | Debug-Go | Execute a program until the next breakpoint |
| Shift F-5 |  | Debug-Stop debugging | To stop debugging a program. It will stop executing the program |
| F-10 |  | Debug-StepOver | Go to the next statement |
| F-11 |  | Debug-Step Into | Go inside a function |
| Shift F-11 |  | Debug – Step Out | Come out of the function |
|  |  | Debug - Run to cursor | Execute all statements till the statement on which the cursor is placed or until the next breakpoint |
| Alt -3 |  | Debug-Windows-Watch | Show the window where only the variables in scope are shown |
| Alt-4 |  | Debug-Windows-Variables | Show the window in which you can type a variable name to see its value |
| Alt-7 |  | debug-windows-call stack | You can see the activation of stack of functions here |

**TASK 2:**

Given two integers x and y, find their sum using pointers.

**TASK 3:**

Given two integers x and y, swap their values using pointers.

**TASK 4:**

1. Declare an array of elements “arr[5]”.
2. Declare a pointer variable “ptr”.
3. Display the address of each element using array.
4. Display the address of each element using pointer.

**TASK 5:**

1. Write a C++ program that finds the median of following three integers using their pointers.

const **int** **a**=5;

const **int** **b**=10;

const **int** **c**=12’

Modify your program a little. Try to assign the value **-1** to whichever integer occurs to be the median. The task is to be done through pointers.

**TASK 6:**

Create a float array **InArr** of size 10 and another float array **ResArr** of size 9. Point a pointer **myptr** to InArr. Now perform the operation ResArr[i] = InArr[i] + InArr[i+1]. Once this operation is completed, point myptr to ResArr.   
In case you are unable to follow given instructions, figure out the issue and its solution. You should be able to explain the phenomenon that caused the problem.