

Objectives:

At the end of the class the students should be able to:

- Use 1D arrays in C programs.

Exercise 1

Following is a sample C program to input five integer numbers from the keyboard and store them within an array called **numbers**. The array elements should be displayed later.

```
#include <stdio.h>
int main(void)
{
    int numbers[5] = {0};
    int i;

    for(i = 0; i < 5; i++) //Store integer values within the array
    {
        printf("Enter integer value : ");
        scanf("%d", &numbers[i]);
    }

    printf("\nArray Elements : ");

    for(i = 0; i < 5; i++) //Display array element
    {
        printf("%d\t", numbers[i]);
    }

    return 0;
}
```

- Type the given C program in Dev C++.
- Compile and run the C program.
- Set a break point at the second statement in the main program.
- Using debugging option, add watches to the array and the variable.
- Using next line button, execute remaining statements and check how the array elements are changed while taking user inputs.

IT1010 – Introduction to Programming**Semester 1, 2021****Exercise 2**

Following is a sample C program that uses a single-subscripted array of size 10 and the array is initialized with set of integer values.

25, 32, 45, 2, 13, 9, 6, 10, 17, 4

The program reads a **search key** (integer value) from the keyboard and display the location within the array, if the search key is found. If not, display "Value not found".

```
#include <stdio.h>
int main(void)
{
    int values[10] = {25, 32, 45, 2, 13, 9, 6, 10, 17, 4};
    int key, i;

    printf("Enter search key : ");
    scanf("%d", &key);

    for(i = 0; i < 10; i++)
    {
        if(key == values[i])
        {
            printf("Index no. : %d\n", i);
            return -1; //Terminate the program execution
        }
    }

    printf("Value not found\n");

    return 0;
}
```

- i) Type the given C program in Dev C++.
- ii) Compile and run the C program.
- iii) Set a break point at the second statement in the main program.
- iv) Using debugging option, add watches to the array and the variables.
- v) Using next line button, execute remaining statements and see how the C program runs when taking different search keys as the user input.

Exercise 3

- i) Write a C program to do the following.
 - a) Create an array called ***data*** to store the monthly average rainfall for a period of one year.
 - b) Input the values from the keyboard and store them in the array.
 - c) Find the month with the maximum rainfall
 - d) Find the month with the minimum rainfall
- ii) Compile and run the program.
- iii) Set a break point at a suitable statement of the main program.
- iv) Using debugging option, add watches to the declared variables and arrays.
- v) Using next line button, execute remaining statements see how the C program runs.