U3 Arrays
(1D and 2D)

Mr. J. Mishra
MGCUB, INDIA

Objectives

Introduction

Declaration and
Initialization

Manipulation

Operations on
Arrays

Array Example

Exercise

References

Arrays-I
Arrays (1D and 2D)

Mr. Joynath Mishra
Assistant Professor (Guest)
Department of Computer Science and Information Technology

Mahatma Gandhi Central University
Bihar, INDIA

May 31, 2020

Outline

U3 Arrays
(1D and 2D)

Mr. J. Mishra
MGCUB, INDIA

Objectives

Introduction

Declaration and
Initialization

Manipulation

Operations on
Arrays

Array Example

Exercise

References

Objectives

- Concept of array
- Study on declaration and initialization of array
- Operations on array
- Study on 1D and 2D array

Introduction

- Collection of similar type of primitive data type
- Addressed by a common variable name
- Homogeneous continous memory location
- Array data types: char, int, float, double
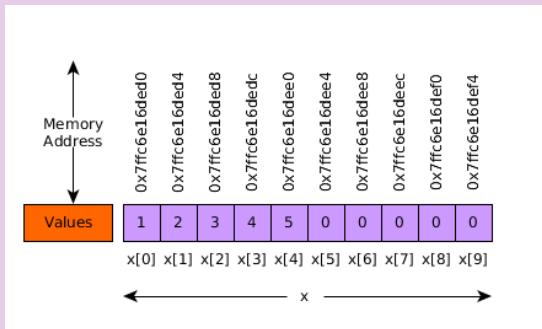- Randomly accessed elements by indices (subscript value starts from 0)

Figure 1

Declaration and Initialization

One Dimensional Array

Declaration and Initialization

```
1   //Array declaration
2   char ch[10];
3   int x[10];
4   float f[10];
5   //Initialization
6   x[10] = {1,2,3,4,5};
7   ch[10] = {'M', 'G', 'C', 'U'};
8   f[10] = {3.2, 5.3, 1.2};
9
10  //Declaration and Initialization
11  int x[10] = {1,2,3,4,5};
12  char ch[10] = {'M', 'G', 'C', 'U'};
13  float f[10] = {3.2, 5.3, 1.2};
14
15  //Dynamic Initialization (without size)
16  int x[] = {1,2,3,4,5};
17  char ch[] = {'M', 'G', 'C', 'U'};
18  float f[] = {3.2, 5.3, 1.2};
```

Graphical Representation

| M | G | C | U | NULL | | | | | |

c[0] c[1] c[2] c[3] c[4] c[5] c[6] c[7] c[8] c[9]

c

Figure 2

| 1 | 2 | 3 | 4 | 5 | 0 | | | | |

x[0] x[1] x[2] x[3] x[4] x[5] x[6] x[7] x[8] x[9]

x

Figure 3

| 3.2 | 5.3 | 1.2 | 0.0 | | | | | | |

f[0] f[1] f[2] f[3] f[4] f[5] f[6] f[7] f[8] f[9]

f
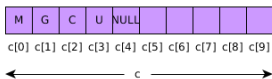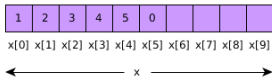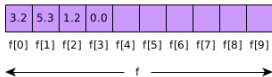
Two Dimensional Array

- Two dimensional array is stored in row-column format
- First indices indicate row and second indices indicate column
- Similarly, multi-dimensional arrays are possible

Declaration and Initialization

```
1  //Declaration and Initialization
2  int x[2][3] = {{1, 3, 0}, {−1, 5, 9}};
3  int x[][3] = {{1, 3, 0}, {−1, 5, 9}};
4  int x[2][3] = {1, 3, 0, −1, 5, 9};
5
6
7  //Initialization with missing values
8  int y[5][3] = {1, 2, 3, 4, 5, 6, 7};
```

Graphical Representation

y[0][0]	y[0][1]	y[0][2]
y[1][0]	y[1][1]	y[1][2]
y[2][0]	y[2][1]	y[2][2]
y[3][0]	y[3][1]	y[3][2]
y[4][0]	y[4][1]	y[4][2]

◀━━━ y[5][3] ━━━▶

Figure 6

1	2	3
4	5	6
7	0	0
0	0	0
0	0	0

◀━━ y[5][3] ━━▶

Figure 7

1	3	0
-1	5	9

◀━x[2][3]━▶

Figure 5

Manipulation

One and Two Dimensional Array

Array Element Manipulation

```c
#include<stdio.h>
int main()
{
    int x[10] = {1, 2, 3, 4, 5};

    printf("Entered array is:");
    for(int i=0; x[i]!=0; i++)
        printf("\t%d", x[i]);


    //Print stored value at x[2]
    printf("\nThird value of x :\t%d", x[2]);


    //Sum of two elements
    int sum = x[1] + x[3];
    printf("\nSum of two elements:\t%d", sum);


    //Storing value in array
    x[3] = 24;
    printf("\nNew value of x[3]:\t%d", x[3]);

    printf("\n");
    printf("\nAddress of array: \t%p", &x);
    printf("\nStarting address of array: \t%p", &x[0]);
    printf("\nLast address of array: \t%p", &x[9]);

    return 0;
}
```

Output

```
Entered array is: 1 2 3 4 5
Third value of x : 3
Sum of two elements: 6
New value of x[3]: 24


Address of array: 0x7ffc6e16ded0
Starting address of array: 0x7ffc6e16ded0
Last address of array: 0x7ffc6e16def4
```

U3 Arrays
(1D and 2D)

Mr. J. Mishra
MGCUB, INDIA

Objectives

Introduction

Declaration and
Initialization

Manipulation

Operations on
Arrays

Array Example

Exercise

References

Operations on Arrays

Traversal
- Processing each element in the array or list

Search
- Find a location and value with given key in array
- Algorithms: linear search, binary search

Insertion
- Adding a new value to array at start, middle, end location

Deletion
- Removing a value from array at start, middle, end loaction

Sorting
- Arranging elements in ascending or descending order
- Algorithms: Insertion, Bubble, Quick sort

Merging
- Compibing two list or arrays

Average Temperature (1D Array)

```c
1  /*Average temperature of a week*/
2  #include<stdio.h>
3  int main()
4  {
5    int i;
6    float average, temp[7], total;
7    printf("Please enter seven days temperature:\t");
8
9    for(i=0;i<7;i++)
10       scanf("%f", &temp[i]);
11
12   for(i=0;i<7;i++)
13       total = total + temp[i];
14
15   average = total/7;
16
17   printf("Seven days average temperature: %f", average);
18   return 0;
19 }
```

Output

Please enter seven days temperature: 12 12 13 22 12 13 14
Seven days average temperature: 14.000000

Array Example

Example 2

Searching largest and smallest value (1D Array)

```
1  /*Find largest and smallest value*/
2  #include<stdio.h>
3  #define MAX 20
4  int main()
5  {
6      int a[MAX],i,n,large,small;
7      printf("How many elements:");
8      scanf("%d",&n);
9      printf("Enter the Array:");
10
11     for(i=0;i<n;++i)
12         scanf("%d",&a[i]);
13
14     large=small=a[0];
15     for(i=1;i<n;++i)
16     {
17         if(a[i]>large)
18             large=a[i];
19         if(a[i]<small)
20             small=a[i];
21     }
22
23     printf("The largest element is %d",large);
24     printf("\nThe smallest element is %d",small);
25
26     return 0;
27 }
```

Output

How many elements:5
Enter the Array:3 7 1 9 3
The largest element is 9
The smallest element is 1

U3 Arrays (1D and 2D)

Mr. J. Mishra MGCUB, INDIA

Objectives

Introduction

Declaration and Initialization

Manipulation

Operations on Arrays

Array Example

Exercise

References

Matrix Addition (2D Array)

```
1   #include <stdio.h>   /* Matrix Addition*/
2   int main()
3   {
4       int r, c, a[50][50], b[50][50], sum[50][50], i, j;
5       printf("Enter number of rows and column: \t");
6       scanf("%d%d", &r, &c);
7       printf("Enter elements of 1st matrix:\n");
8       for (i = 0; i < r; ++i)
9           for (j = 0; j < c; ++j)
10              scanf("%d", &a[i][j]);
11
12      printf("Enter elements of 2nd matrix:\n");
13      for (i = 0; i < r; ++i)
14          for (j = 0; j < c; ++j)
15              scanf("%d", &b[i][j]);
16
17      for (i = 0; i < r; ++i)      // adding two matrices
18          for (j = 0; j < c; ++j)
19              sum[i][j] = a[i][j] + b[i][j];
20
21      printf("\nSum of two matrices: \n");
22      for (i = 0; i < r; ++i)      // printing the result
23      {
24          for (j = 0; j < c; ++j)
25              printf("%d    ", sum[i][j]);
26          printf("\n\n");
27      }
28      return 0;
29  }
```

Output

Enter number of rows and column: 3 3
Enter elements of 1st matrix:
1 2 3
4 5 6
7 8 9
Enter elements of 2nd matrix:
9 8 7
6 5 4
3 2 1

Sum of two matrices:
10 10 10

10 10 10

10 10 10

Array Example

Example 4

U3 Arrays
(1D and 2D)

Mr. J. Mishra
MGCUB, INDIA

Objectives

Introduction

Declaration and
Initialization

Manipulation

Operations on
Arrays

Array Example

Exercise

References

Transpose Matrix (2D Array)

```c
#include <stdio.h>  /* Transpose Matrix*/
int main()
{
    int r, c, a[50][50], i, j;
    printf("Enter number of rows and column: \t");
    scanf("%d%d", &r, &c);
    printf("Enter elements of 1st matrix:\n");
    for (i = 0; i < r; ++i)
        for (j = 0; j < c; ++j)
            scanf("%d", &a[i][j]);

    printf("\nTranspose matrix: \n");
    for (i = 0; i < c; ++i)
    {
        for (j = 0; j < r; ++j)
            printf("%d    ", a[j][i]);
        printf("\n");
    }
    return 0;
}
```

Output

Enter number of rows and column: 2 4
Enter elements of 1st matrix:
1 2 3 4
5 6 7 8

Transpose matrix:
1 5
2 6
3 7
4 8

Exercise

- Write a program to find the second largest value from an array.
- Write a program to find the substraction of two matrix.
- Write a program to find the sum of two three dimensional vectors(i,j,k).
- Write a program to do a scalar multiplication to a matrix.
- Write a program to print an array in reverse order.
- Write a program to count the prime number in an array.

📄 S. Jain, *Programming and Problem Solving through C Language Design*. BPB Publications, 2003.

📄 K. Yashwant, "Let us c," *Array and pointers, 7th edition, BPB publication*, 1999.

📄 ——, *Test Your C Skills*. BPB publication, 2005.

Get in touch via...

+91 9046174189

jaynath4025@gmail.com

Thank You...