ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

Array

Department of Information System

SoICT, HUST

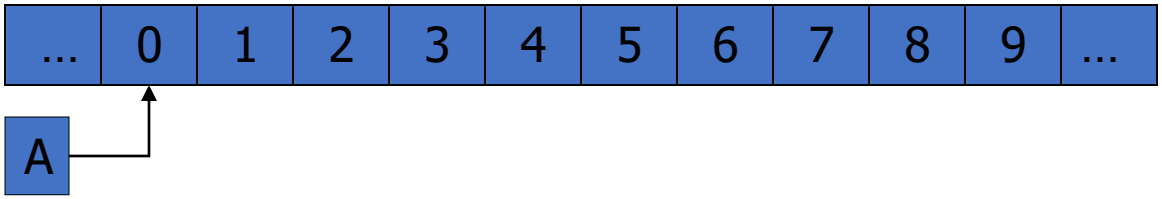
Array

- Array can declare any data type
  - Example, `int A[10]` creates an array of 10 integers
- The total elements of an array is a constant

Memory of an array

- A group of continuous memory locations used to store a series of related values
- All values have the same type
- Array's name is the address of the first element in the memory.
- Example:

  int A[10];



- Individual elements of an array are accessed via an integer index.
- Element indices start at 0 (index of the first element).

Program

- Read in a series of integer numbers (10 numbers at maximum) to store in an array and print the array in reverse order.

Program

- Read in a series of integer numbers (10 numbers at maximum) to store in an array and print the array in reverse order.

```
#include <stdio.h>
int main(void)
{
   int i, n, A[10];
   printf("Nhap so phan tu trong day (n<=10):");
   scanf("%d",&n);

   printf("Nhap cac phan tu trong day:\n");
   for(i=0; i<n; i++) {
      printf("Phan tu thu %d:", i+1);
      scanf("%d",&A[i]);
   }

   printf("Day so sau khi dao lai:\n");
   for(i=n-1; i>=0; i--)
      printf("%5d",A[i]);

   return 0;
}
```

Exercise

- Read in a series of integer numbers to store in an array and calculate their sum.
- Find min, max elements of the array.

Initialize an array

- Arrays may be initialized with a list of suitable values
  - int a[3] = {1, 2, 3};
- The number of elements that are initialized cannot be more than array's size.
  - But it can be smaller
  - At that time, the other elements are initiated with 0.
- Array's size can be induced from the number of elements whose value is initialized.

```
int A[8] = {2, 4, 6, 8, 10, 12, 14, 16};
```

or

```
int A[] = {2, 4, 6, 8, 10, 12, 14, 16};
```

Example

```
int    month;
int table[12] = { 30, 40, 45, 95, 130, 220, 210,
  185, 135, 80, 40, 45 };

printf("Enter a month: ");
scanf("%d", &month);

if (1 <= month && month <= 12)
   printf("Average rainfall for month %d is %d
  mm.\n", month, table[month-1]);
else
   printf("Invalid month\n");
```

Sorting elements in an array

- We iterate the elements from left to right in the array to exchange them with the smallest element on the right side.
  - A[i] is the element to be exchanged
  - A[j] is an element on the right side of A[i]
  - A[i] must be exchanged with A[j] if A[i] < A[j]
  - To exchange two elements, use a temporary memory

Program

```
#include <stdio.h>
int main(void)
{
   int A[10], n, i, j, tmp;
   /* enter numbers into array */
   for(i=0; i<n-1; i++)
     for(j=i+1; j<n; j++)
       if (a[i]<a[j]) {
         tmp = A[i];
         A[i] = A[j];
         A[j] = tmp;
       }
   printf("Achieved array:\n");
   for(i=0; i<n-1; i++)
      printf("%5d", A[i]);
   return 0;
}
```

Array index

- Arrays have a fixed size.
- There is no built-in way of checking if the supplied index is within range.
- We must check for valid indices ourselves.

Example:

```
int i, n=5;
int A[n];
int B[5];
for (i=1; i<=n; i++)
    B[i] = 0;
```

Array's size is a constant

Error: index is out of array's scope

Passing arrays to a function

- The array is passed as an array of unspecified size (int array[]) and the total number of elements.
  - `f (int array[], int n)`
- Array's elements can be changed

Passing arrays to a function

```
#include <stdio.h>

void inputArray(int array[], int num)
{
    int i;
    for(i=0; i<num; i++) {
        printf("i-th element %d:", i+1);
        scanf("%d", &array[i]);
    }
}

void reverse(int array[], int num)
{
    int i;
    for(i=num-1; i>=0; i--)
        printf("%5d", array[i]);
}
```

Passing arrays to a function

```
int main(void)
{
    int n, A[10];
    printf("Enter number of array elements(n<=10):");
    scanf("%d",&n);

    printf("Enter each element:\n");
    inputArray(A, n);

    printf("Reverse array:\n");
    reverse(A, n);

    return 0;
}
```

2D array:

- An array's element has a specific data type
- An array can be considered as data type for an array's element.
  - An element of an array can be another array
- A 2D array is: an "array-of-arrays"

Example:

```
int A[5][3];
for (i=0; i<=5; i++)
   for (j=0; j<=3; j++)
      A[i][j] = 10;
```

Example

```
#define  NYEARS   5
#define  NMONTHS  12
int table[NYEARS][NMONTHS] ={
      {30,40,75,95,130,220,210,185,135,80,40,45},
      {25,25,80,75,115,270,200,165, 85, 5,10, 0},
      {35,45,90,80,100,205,135,140,170,75,60,95},
      {30,40,70,70, 90,180,180,210,145,35,85,80},
      {30,35,30,90,150,230,305,295, 60,95,80,30}
};
```

table[i][j] allows to access rainfall in month (j+1) and year (2000+i)

Example

```
printf("Enter a year: ");
sscanf("%d", &year);
if (year < 2000 && year > 2004) {
  printf("Year between 2000 and 2004\n");
  return 1;
}
printf("Enter a month: ");
scanf("%d", &month);
if (month < 1 && month > 12) {
  printf("Invalid month\n");
  return 1;
}
printf("Average rainfall is mm.\n", table[year-
  2000][month-1]);
```

Example

```
#include <stdio.h>

void main()
{
   /* bảng cửu chương cho cả số 0 */
   int cuuchuong[10][10];
   int i, j;

   /* tạo giá trị cho bảng cửu chương */
   for (i=0; i<=9; i++)
      for (j=0; j<=9; j++)
         cuuchuong[i][j] = i*j;

   printf("Nhap hai so cua bang cuu chuong\n");
   printf("So 1: "); scanf("%d", &i);
   printf("So 2: "); scanf("%d", &j);

   printf("Giá trị trong bang la %d", cuuchuong[i][j]);
}
```

Exercise 1

- Given function's prototypes:

```
int Enter_ Number_Element(int a[]);
int Find_Max(int a[], int n);
```

- The function **`Enter_ Number_Element`**() allows to input the number of elements and their values. It returns the number of elements of the array. The function Find_Max() find the maximum value in the input array.
- Write functions' definition and the main function that allows to input two different arrays and find the maximum value in both arrays

Exercise 2

Solve the following set of equations:

a11*x + a12*y = c1

a21*x + a22*y = c2

1. Input values for the 2D array **a**.
2. Input values for the array **c**.
3. Compute and display values for x and y.

Exercise 3

- Read in a series of integer numbers to store in a 2D array
- Sort elements of the array in increasing order by two ways:
  - Using an extra array
  - Without using an extra array

ĐẠI HỌC BÁCH KHOA

25 YEARS ANNIVERSARY SOICT

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY

Thank you for your attentions!

soict.hust.edu.vn/ fb.com/groups/soict