

Lec 9

Arrays :-

What's an array in C/C++?

An array is essentially a collection of elements.

هي عبارة عن مجموعة من العناصر

The data types for all elements must be the same and store at the contiguous memory location

كل العناصر من نفس نوع الداتا تألي سواء كانت Float, Char, Int
والعناصر تكون متجاورة في الذاكرة ← يعني العناصر ورا بعضها

Each element of the array is individually referenced by using an index

رقم
كل عنصر من العناصر له index خاص به يتم الوصول للعنصر من خلاله

We can easily access the element using an index in square brackets.

يتم الوصول للعنصر من خلال الindex كـ [index]

The index of the array always starts with 0

الindex الخاص بال array أول عنصر يبدأ بـ index ← 0

Array Zero based

Array Syntax (Declaration of Array in c/c++)

Data type Array Name [Array size];

Ex:-

unsigned int std_IDS [5];

كدا انا انا انا array ← بتكون من 5 عناصر من نوع unsigned int كل عنصر منهم 4 bytes ← array كذا حجمها $4 \times 5 = 20$ bytes

لو عايز انك تاكل اسم ال array وتاخذ ال sizeof operator

printf("size = %i \n", sizeof(std_IDS));

out put → size = 20

* لو عايز اعمل initialization ال array

① definition

index → 0 1 2 3 4

unsigned int std_IDS[5] = {0x55, 0x33, 0x21, 0x2, 0x2};

first element

last element

أول index بتدي 0 وآخر index ← $n-1$ حيث n عدد عناصر ال array ← size

② عمل write لكل عنصر لوحده باستخدام ال index بعد معرفت ال array definition

std_IDS[0] = 0x22;

كدا العنصر الأول متخزن فيه 0x22

std_IDS[1] = 0x33;

والثاني 0x33

لو عاينز أقرأ عنصر في الأري
 printf("Value = 0x%X \n", std-Ids[0]);
 لـ index الخاص بالعنصر

unsigned int arr[5] = {0x99}

لو أنا كتب حاجة زي كذا قيمة واحدة بس ← هياخذها وتخزنها
 في أول عنصر فقط وباقي العناصر قيمتها هتبقى Zero ← الـ dam
 لو أنا هعرف الـ array في الجوال Global لو هعرفها في local
 ممكن ياقى العناصر تبقى Garbage Value وممكن تبقى Zero

Note

Size and type of an array "static configuration"

That mean cannot change once it is declared.

- The first element is the lowest address and the last element is at the highest address in an array

unsigned int Array[5];

Address

Value

Array[0]

0x00000000

0x30

Array[1]

0x00000004

0x31

Array[2]

0x00000008

0x32

Array[3]

0x0000000c

0x33

Array[4]

0x00000010

0x34

Initialization of array in c (Designated Initializers)

The C99 introduces a new mechanism to initialize the elements of the array

index → 0 1 2 3

unsigned int Array[20] = {1, 2, 3, [15] = 0x66, 5, [13] = 0x77};

يمكن اعل write لعناصر ال array بالطريقة دي وهو ميخزن
 1 ← index ← 0 2 ← index ← 1 3 ← index ← 2 5 ← index ← 3 0x66 ← index ← 15 0x77 ← index ← 13

الطريقة دي depend on compiler يعني يفضل مستخدمهاش

Boundary check in array

علا C عندي مشكلة مع boundary check يعني محدش جردو للأرأى يعني لو هفترض المثال دا مثلا

unsigned int std_IDS[5] = {0x11, 0x22, 0x33, 0x44, 0x55}

و جيت علت حاجه بالشكل دا

std_IDS[100] = 0x99;

ال C بتقبل حاجه قبيحة كذا ولو جيت طبعتا طباع قيمة ال index ← 100

printf("value = 0x %X\n", std_IDS[100]);

output → value = 0x99

another example :-

```
unsigned int std_IDS[5] = {0x11, 0x22, 0x33, 0x44, 0x55};
```

```
unsigned int NumberOne = 0x88;
```

```
printf("NumberOne = 0x%X \n", NumberOne);
```

```
std_IDS[5] = 0x99;
```

```
printf("value = 0x%X \n", std_IDS[5]);
```

```
printf("NumberOne = 0x%X \n", NumberOne);
```

out put :-

NumberOne = 0x88

Value = 0x99

NumberOne = 0x99

* الحصول هنا وان NumberOne لا تزن في الميموري
بعد الاقراى وطول واناجيت على access
لـ index من موجود في حدود الـ size بتاع الـ array
فالكميل خذ القيمة بتاع الـ index الـ ابراحود
الـ array وتخزنها في المتغير الـ المتجز بعد
الـ array فغسان كنا بنقول ان الـ array
له امتش bounding check

Note :-

اسم الـ array اوحده هو عبارة عن عنوان اول عنصر في الـ array

```
printf("Address = 0x%X \n", &std_IDS[0]);
```

```
printf("Address = 0x%X \n", std_IDS);
```

الـ انيت Printf هيطلعو نفس الـ result

* لو عايز اروح لذي عنصر في ال array

$$ARR[x] = \text{عنوان أول عنصر} + (\text{index} * \text{حجم العنصر الواحد})$$

↑ index

هذه unsigned int

$$ARR[0] = 0 + (0 * 4) = 0$$

هتفرض انه عنوان صفر

$$ARR[3] = 0 + (3 * 4) = 12$$

* ال address بتاع ال index ← 3 ← 12

على افتراض بان عنوان أول عنصر 0

* ذهن الوصول لذي عنصر في ال array ثابت عنوان كل مرة بتطبق في المعادلة الا فوق عنوان كدا بقول ان ال array اوسع أنواع ال data storage أو data structure

* using Array with for loop :-

بستخدم ال for مع ال array عنوان عارف البداية و النهاية ال array
* ممكن استخدمها مع printf أو scanf

```
unsigned int cnt = 0;
for (cnt = 0; cnt < size of array; cnt++) {
    printf("value of index %i is : %i", cnt, arr[cnt]);
}
```

← read

```
unsigned int cnt = 0;
for (cnt = 0; cnt < size of array; cnt++) {
    scanf("%i", &arr[cnt]);
}
```

← write

* لو عايز اعرف عدد العناصر الـ في الـ array
فعمل متغير وباصيله ناتج المعادلة دي -

`unsigned int my_array[5];`

`unsigned int my_array_size = sizeof(my_array) / sizeof(my_array[0]);`

← حجم أول عنصر في الـ array
← حجم العناصر كلها الـ في الـ array

$$= 20 \div 4 = 5$$

* كل الشرح الـ في الـ one dimensional array يعني قولين بس بعد
اسم الـ array بالشكل دا [] فيه أنواع أخرى Two & Three
بس قبل هشرح ههم هشرح باستخدام الـ one مع الـ string

*** using one dimensional array with string:-**

* في لغة الـ C هفيش داتا تايب string زي الـ C++ هتلا أو اللغات
التانية يعني لو عندي نص وعايز اخزنه هفيش داتا تايب تعده
اللام مع غير لان اهل الـ array of char بالشكل دا

`char Array[] = "Mostaf Alaa Elsisy";`

* دا معناه ان هيجزلي مكان في الـ memory لكل حرف فيها أو مسافة
أو رمز بـ 1 byte

name of array

* لو عايز اطبع الـ string دا ؟

`printf("%s \n", Array);`

* هنا هيرجع الـ address بتاع أول عنصر في الـ array وبطبيعة حرف حرف لحـ
ليوصل لآخر عنصر

* في طريقة ثانية لوعاين اخرون نص في الذاكرة بالشكل دالـ

```
char Name[] = {'M', 'o', 's', 't', 'a', 'f', 'a'}
```

نحتم كل حرف في سينجل route لومده وبين كل حرف والثاني comma

* لا يجب عمل array لعمل حساب لعنصر زيادة character

عمل كذا ليه ؟! عشان كل اخر string لازم يكون في حاجة واسمها Null termination في الينتهي ال string في الميموري

طب لومعناش حساب لآخر عنصر ؟! هيجعل عندي مشاكل في الميموري

* لوعاين المستقبل string من ال User
 char Message[10] = "Most" في array هفترض ابعده عندي

scanf ("%s", Message)

* هنا في ال scanf بيأخذ ليها اسم ال array بس من غير %s

* في ال scanf مش بتقبل مسافات يعني لو مدخله tt tt
 هيخزن tt بس والباقي مش هيخزن عشان اهل حاجة زي

كدا هستخدم builtin function
 gets(Message) ← name of array only

هتخزن المسافات عادي ←

Example:- Security Password

```

unsigned char Pass[5] = {'a', 'b', 'c', '1', '2'};
unsigned char UserPass[5];
unsigned flagPass = 0;
unsigned int index = 0;
printf("Please enter pass : ");
for (index = 0; index < 5; index++) {
    scanf("%c", &UserPass[index]);

```

```

    fflush(stdin); /* clear buffer */
}

```

```

printf("%s", UserPass);

```

* مشكلة في C ← buffering ← %c في scanf لا يتعامل مع الـ Enter كأي Character
 ← char

* حل: استخدام الفانكشن الجاهزة دي

```

fflush(stdin); ← بعد كل scanf و %c

```

```

unsigned int indexPass = 0;
for (indexPass = 0; indexPass < 5; indexPass++) {
    if (Pass[indexPass] != UserPass[indexPass])
    {
        flagPass = 1;
        break;
    }
}

```

```

if (flagPass == 1) {
    printf("Not invalid\n");
} else {
    printf("Welcome\n");
}

```


Two dimensional array :-

unsigned int Numbers[5]; ← one

unsigned int Numbers[3][3]; ← two

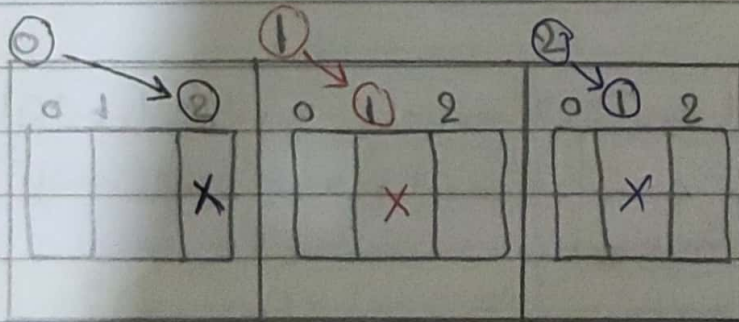
unsigned int Numbers[3][3][3]; ← three

unsigned int Numbers[3][3][3][3]; ← four

* وهكذا كلما زاد الأبعاد dimensionality كل أما size يزيد
كل أما أزور []

unsigned int Numbers[3][3];

* دي كذا عبارة عن Two dimensional ال size بتاعها 3x3
يعني عبارة عن 9 عناصر من نوع unsigned int ويكونو
متخزينين في الذاكرة متجاورين يعني ورا بعض.



* كل عنصر في الأرای الأولی
عبارة عن 3 عناصر
هي الأرای التالیة

ممكن أيضاً أن نأخذها بالشكل دا

		0	1	2
0		00	01	02
1		10	11	12
2		20	21	22

* كذا بهت عبارة عن Matrix مصفوفة

Numbers[3][3]

لـ يبدل على رقم المصفوف
لـ يبدل على رقم الأعمدة

* initialize two dimensional array

①

```
unsigned int Numbers[3][3] = {1, 2, 3, 4, 5, 6, 7, 8, 9};
```

②

```
unsigned int Numbers[3][3] = { {1, 2, 3}, {4, 5, 6}, {7, 8, 9} };
```

③

```
Numbers[0][0] = 1;      * Access لا يمكن الوصول
```

* using loop with two dimensional array

loop ← two for loop مع الـ two dimensions

Nested ← جوا لبعض بالـ another

```
unsigned int index1 = 0, index2 = 0;
```

```
for (index1 = 0; index1 < 3; index1++)
{
```

```
for (index2 = 0; index2 < 3; index2++)
{
```

```
printf("%i \t", Numbers[index1][index2]);      ← for read
```

```
scanf("%i", &Numbers[index1][index2]);      ← for write
```

```
}
```

```
printf("\n");
```

```
}
```


unsigned int [][3] ; → Accepted
 unsigned int [3][] ; → wrong → error

* ينبغي أن أناحدش عدد المصفوف ليس ميفعش محدش عدد الأعمدة

Two
 * ممكن استخدم الـ 2D strings لو أنا عايز اخزن strings

char stdNames [5][30] ;

* هنا معناها أن عندي 5 عناصر كل عنصر فيهم اقدر
 اخزن فيه 30 حرف يعني وليت عندي 5 أسماء كل اسم بيتكون
 من 30 حرف

Ex:-

char Names [5][30] = {"Ahmed", "Mohamed", "Mostafa", "Ali",
 "Mahmoud"} ;

لو عايز اعمل access ← أمبعا مثلا
 for (index1 = 0 ; index1 < 5 ; index1++)

{
 printf ("%s \n", Names[index1]) ;
 }

* هنا لما جيت الطبع الـ 2D استعملت الاسم وأول الـ 2D dimensional
 مع %s عشان نطبع

Function with Array

1- with one dimensional 1D

How to Pass an array as a Parameter in C?

- 1) Parameters as a Pointer. → in Pointer Part
- 2) Parameters as a sized array.
- 3) Parameters as an un-sized array.

unsigned int stdIds[5] = {11, 12, 23, 24, 25};

لو مايز اطلع ال array بقا تلتش

```
void PrintIds(unsigned int stdIdsArray[],
              unsigned int stdIdsLen)
```

{

unsigned index = 0;

for (index = 0; index < stdIdsLen; index++)

{

printf("%i \t", stdIdsArray[index]);

}

printf("\n");

}

* أي فانتكت هتعمل أي أوبريشن على أي أراي على الأقل

الفانتكت لازم تكون عارفه ← الأراي ← size أو length

PrintIds(stdIds, 5); ← Call

اسم ال array

Length

address بياح ال array

لو هتس عايز ال use بعدل في البارامتر بتاع الفانكشن سواء كان ال length بتاعها أو اعدل في أي عنصر من عناصرها

← هتستخدم كلمة const قبل الداتا تايب

void PrintSTIDs(const unsigned int STIDsArray[],
const unsigned int STIDsLen);

و في ال definition نفس الكلام ← هتستخدم كلمة const مكان مقدريش اعدل على البارامتر ← مهمة جداً

* ممكن وأنا بياخد ال array في الفانكشن احدد عدد العناصر
بس يفضل إنك هتحددش وقباصي كمتغير

2. With two dimensional array

* وأنا بياخد ال 2D ← لازم على الأقل احدد عدد عناصر
الأسطر أو ال dimensional الثاني غير كذا هيعمل البرور

* نفس فكرة ال 1D وكل حاجة بس أهم بوييت إن احدد عدد
عناصر الأسطر وممكن محددين الصفوف وأباصيها كمتغير كبرامتر
في الفانكشن