

# جامعة المنارة

مقرر برمجة 2

د. هلا نصار

م. أوشين داود

جامعة  
المنارة

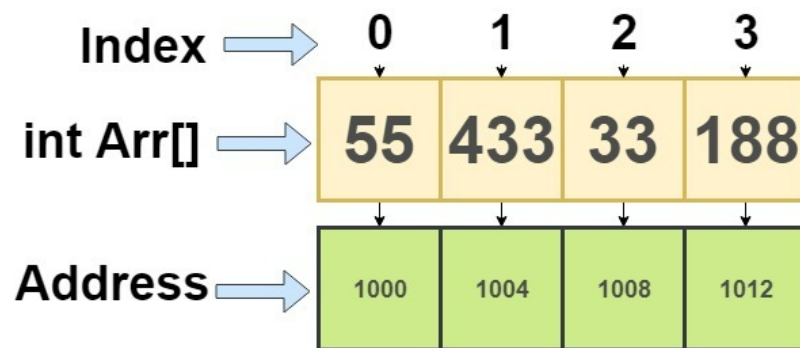
HANARA UNIVERSITY

محاضرات الأسبوع 2

الفصل الثاني-2020/2021

## Pointers and 1-D arrays

ليكن لدينا مصفوفة أحادية البعد :  
إن اسم المصفوفة يحمل عنوان اول عنصر فيها ومنه يمكن اسناد اسم المصفوفة الى مؤشر، شرط أن يكون المؤشر يشير لنوع مماثل لنوع المصفوفة.



ملاحظة: في المصفوفات انتبه الى الفرق بين  
الفهرس والقيمة والعنوان

HAHARA UNIVERSITY

لدينا مصفوفة أحادية a من النوع int ومؤشر إلى النوع نفسه.

```
#include <iostream>
using namespace std;
int main()
{ int a[5] = {30,5,66,12,0};
int *p;
p=a;
cout <<"p= " << p << endl;
cout <<"a= " << a << endl;
cout <<"&a= " << &a << endl;
cout <<"&a[0]= " << &a[0] << endl;

cout <<"*p= " << *p << endl;
cout <<"a[0]= " << a[0] << endl;
return 0;
}
```

يحمل اسم المصفوفة عنوان أول عنصر فيها. عند اسناد اسم المصفوفة إلى مؤشر فيمكن طباعة العنوان من خلال:  
طباعة اسم المؤشر  
أو اسم المصفوفة مباشرة  
أو من خلال استخدام & مع اسم المصفوفة  
أو استخدام & مع العنصر الأول من المصفوفة.

The image shows a C++ IDE window titled 'main.cpp' with the following code:

```
1 #include <iostream>
2 using namespace std;
3 int main()
4 { int a[5] = {30,5,66,12,0};
5   int *p;
6   p=& a;
7   cout <<"p= " << p<<endl;
8   cout <<"*p= " <<*p<<endl;
9   return 0;
10 }
11
```

To the right of the code, there is Arabic text: لا يمكن استخدام معامل & مع اسم المصفوفة عند اسناد قيمته الى مؤشر

Below the code editor, the 'Logs & others' panel shows a 'Build messages' window with the following error message:

```
=== Build: Debug in p2 (compiler: GNU GCC Compiler) ===
In function 'int main()':
C:\Users\HelpD... 6 error: cannot convert 'int (*)[5]' to 'int*' in assignment
=== Build failed: 1 error(s), 0 warning(s) (0 minute(s), 0 second(s)) ===
```

## طباعة عناصر مصفوفة باستخدام المؤشرات

يمكننا استخدام العمليات الحسابية لطباعة العناصر من العنصر الأول في المصفوفة الى العنصر الأخير مع موقع كل عنصر كما هو موضح في المثال التالي :

```
#include <iostream>
using namespace std;
int main() {
int a[5] = {30,5,66,12,0};
int *p ;
p = a;
//Array values using pointer
cout << "the value of the first element " << *p << "at location:" << p << endl ;
cout << "the value of the second element " << *(p+1) << "at location:" << p+1 << endl;
cout << "the value of the third element " << *(p+2) << "at location:" << p+2 << endl ;
cout << "the value of the fourth element " << *(p+3) << "at location:" << p+3 << endl ;
cout << "the value of the fifth element " << *(p+4) << "at location:" << p+4 << endl ;
cout << "the value of the first element " << *p << "at location:" << p << endl ;
return 0;
}
```

يمكن استخدام الحلقات والعمليات الحسابية لطباعة قيم العناصر من العنصر الأول في المصفوفة الى العنصر الأخير مع العنوان كما هو موضح في المثال التالي

```
#include <iostream>
using namespace std;
int main() {
int a[5] = {30,5,66,12,0};
int *p ;
p = a;
//Array values using pointer
for (int i = 0 ; i<5 ; i++)
    {cout << * (p+i)<< "  " << p+i << endl ;}
cout << "the value of the first element " << *p << "at location:" << p << endl ;
return 0;
}
```

## انتبه الى الفرق بين البرنامجين التاليين:

```
#include <iostream>
using namespace std;
int main()
{int a [5] = {55,66,77,11,22};
int *p = a ;
for (int i =0 ; i< 5 ; i++)
cout << *(p++)<< endl ;
return 0; }
```



```
#include <iostream>
using namespace std;
int main()
{int a [5] = {55,66,77,11,22};
int *p = a ;
for (int i =0 ; i< 5 ; i++)
cout << (*p)++<< endl ;
return 0; }
```

يمكن استخدام الحلقات والعمليات الحسابية للانتقال من العنصر الأول في المصفوفة الى باقي العناصر وطباعتها مع العنوان الموافق لكل عنصر كما هو موضح في المثال التالي :

```
#include <iostream>
using namespace std;
int main() {
int a[5] = {30,5,66,12,0};
int *p ;
p = a;
//Array values using pointer
for (int i = 0 ; i<5 ; i++)
{cout << * p<< " " << p << endl ;
p++;}
cout << "the value of the element " << *p<< "at location:" << p<< endl ;
return 0;
}
```



لاحظ الفروقات بين البرامج السابقة والاختلاف طريقة طباعة قيم العناصر باستخدام المؤشر وعند نقل المؤشر بشكل متتالي على عناصر المصفوفة.

```
#include <iostream>
using namespace std;
int main() {
int a[5] = {30,5,66,12,0};
int *p ;
p = a;
//Array values using pointer
for (int i = 0 ; i<5 ; i++)
    {cout << * (a+i) << "  " << p << endl ;}
cout << "the value of the element " << *p << "at location:" << p << endl ;
return 0;
}
```

اكتب برنامج لطباعة عناصر مصفوفة وعناوينها ابتداء من اخر عنصر إلى أول عنصر فيها باستخدام مؤشر ينتقل على عناصر المصفوفة.

```
#include <iostream>
using namespace std;
int main() {
int a[5] = {30,5,66,12,0};
int *p ;
p = &a[4];
for (int i = 4 ; i>=0 ; i--)
    {cout << * p<< " " << p << endl ;
    p--;}
return 0;
}
```

HANARA UNIVERSITY

## Pointers and 1-D arrays :

تعلمنا كيف نقوم بتعريف مؤشر الى اول عنصر من المصفوفة وفق الصيغة :

```
int arr[5] = { 1, 2, 3, 4, 5 };
```

```
int *ptr = arr;
```

كيف يتم تعريف مؤشر الى المصفوفة كلها؟  
الصيغة :

```
data_type (*var_name)[size_of_array];
```

مثال :

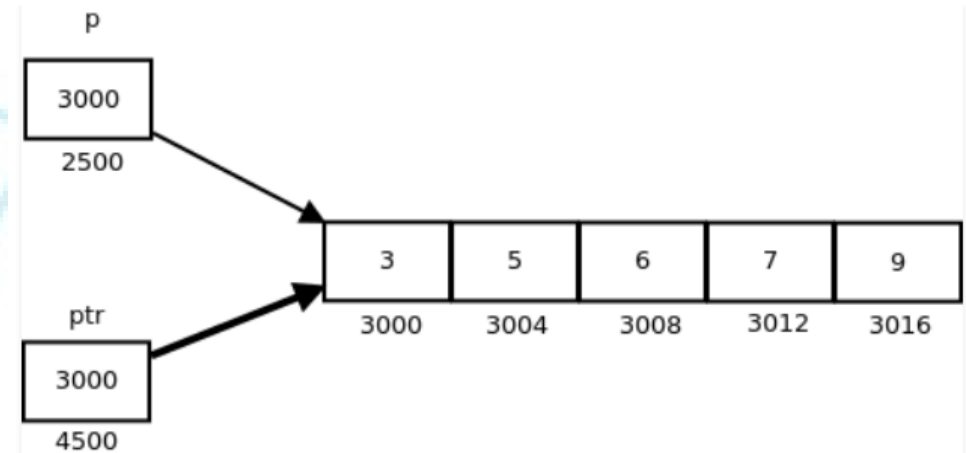
```
// Pointer to an array of 5 integers
```

```
int (*ptr)[5];
```

```
int arr[5];
```

## الاختلاف بين المؤشر الى اول عنصر من المصفوفة والمؤشر الى المصفوفة ككل

```
#include <iostream>
using namespace std;
int main()
{// Pointer to an integer
int *p;
// Pointer to an array of 5 integers
int (*ptr)[5];
int arr[5];
// Points to 0th element of the arr.
p = arr;
// Points to the whole array arr.
ptr = &arr;
cout << "p =" << p << ", ptr = " << ptr << endl;
p++; //the pointer p will be shifted forward by 4 bytes
ptr++; //the pointer ptr will be shifted forward by 20 bytes
cout << "p =" << p << ", ptr = " << ptr << endl;
return 0;
}
```



## QUIZ

قم بتهيئة مصفوفة كما يلي :

```
int a [6] = {55,66,77,11,22,88};
```

ثم طباعة الاعداد من الثاني الى الخامس باستخدام مؤشر p ينتقل على عناصر المصفوفة

```
#include <iostream>
using namespace std;
int main()
{int a [6] = {55,66,77,11,22,88};
int *p = &a[1];
for (int i =1 ; i< 5 ; i++)
{cout << *p<< " " << p<< endl ;
p++;}
return 0; }
```

انتهت تمارين الأسبوع الثاني

جامعة  
المنارة  
HANARA UNIVERSITY