كلية العلوم

القسم : **الرياضيات**

السنة : **الثانية**

المادة : لغات البرمجة٢

المحاضرة : الاولى/عملي/

{{ مكتبة A to Z }}

Facebook Group : مكتبة A to Z

الجمهورية العربية السورية

جامعة طرطوس

كلية العلوم قسم الرياضيات

السنة الثانية

المادة: لغات برمجة2 _ عملي

المحاضرة الأولى

مراجعة المصفوفات أحادية البعد

المصفوفات Arrays

4.1) المصفوفات وحيدة البعد One Dimensional Arrays

المصفوفة: هي مجموعة مواقع الذاكرة المتجاورة ذات اسم واحد والمخصصة لحفظ عناصر المصفوفة من نفس النوع(ثوابت صحيحة أو حقيقية أو محرفية أو مجموعة محارف) ويتم الوصول اليها للكتابة فيها، أو القراءة منها باستخدام دليل المصفوفة index، وهو عدد صحيح موجب يحدد عنوان موقع عنصر المصفوفة وبالتتابع 0,1,2,3,... وهذه الأرقام تسمى بفهرس عنصر المصفوفة. إذا كان اسم المصفوفة a، فإن a[0] هو اسم العنصر الموجود في موقع الذاكرة رقم صفر(المكان الأول) و a[1] هو اسم العنصر الموجود في موقع الذاكرة رقم 1 وهكذا. بصورة عامة فإن عنصر المصفوفة ذو الدليل i يتوضع في موقع الذاكرة رقم i−1، وإذا كانت المصفوفة تحتوي على n عنصر فإن أسماء هذه العناصر ستكون a[0], a[1], ...a[n−1]، ويمكن تصور المصفوفة a المؤلفة من اربع عناصر من النوع الحقيقي احادية الدقة {3.14 , 43.35 , 22.47, 17.27} =a[4] كالآتي:

	0	1	2	3
a	3.14	43.35	22.47	17.27

شكل (4.1) مصفوفة أعداد حقيقية أحادية الدقة

الشكل السابق(4.1) يبين مصفوفة أعداد حقيقية أحادية الدقة (نوعها float) اسمها a مكونة من أربعة عناصر: العنصرالأول a[0] (ذو الدليل الصفري) وقيمته الثابت 3.14 والعنصر a[1] (ذو الدليل1) وقيمته الثابت 43.35 والعنصر a[2] (ذو الدليل2)وقيمته الثابت 22.47 والعنصر a[3] وقيمته الثابت17.27.

4.2) التصريح عن المصفوفات Declaring Arrays

يتم التصريح عن المصفوفة في لغة C++ بالشكل الآتي:

ArrayType arrayName [arraySize];

حيث أن:

ArrayType: نوع عناصر المصفوفة(ثوابت صحيحة أو حقيقية أحادية أو مضاعفة الدقة أو محرفية أو سلسلة محارف).

arrayName: اسم المصفوفة.

arraySize: عدد عناصر المصفوفة (حجمها).

تتطلب لغة C++ القياسية أن يكون حجم المصفوفة arraySize ثابتاً صحيحاً موجباً وبموجب العبارة double a [4]; يتم التصريح عن مصفوفة اسمها a نوعها حقيقي مضاعفة الدقة، عدد عناصرها 4، ويمكن التصريح عن المصفوفة بالشكل التالي ايضاً:

```
const   int size= 4 ;
double  a[ size ];
```

4.3) إدخال عناصر المصفوفة Input The Elements of Array.

يُستخدم دليل المصفوفة لتحديد العنصر المراد إدخال قيمته وتخزينها في ذاكرة الحاسب باستخدام أي من بنى التنفيذ الحلقيloop وكذلك الأمر بالنسبة لعملية إظهار قيمته على الشاشة، وبشكل عام يتم استخدام البنية التكرارية for لتحديد دليل المصفوفة، حيث يجب تنفيذ عملية التمهيد من لوحة المفاتيح أوفي جسم البرنامج، ويتم إدخال قيم عناصر المصفوفة a [4] من لوحة المفاتيح وفق جزء الشفرة الآتية:

```
for (int i=0; i< max  ;  i++)
cin >> a[i];              //أداة إدخال قيم عناصر المصفوفة من لوحة المفاتيح
```

4.4) إظهار عناصر المصفوفة Displaying The Elements of Array

يستخدم دليل المصفوفة لتحديد العنصر المراد إظهار قيمته على الشاشة باستخدام أي من بنى التنفيذ الحلقي فمثلاً لإظهار قيم عناصر المصفوفة نكتب جزء الشفرة الآتي:

```
for (int i=0; i < 4  ;  i++)
cout<< a[i];              //أداة إظهار قيم عناصر المصفوفة
```

مثال 1:

اكتب برنامج يقوم بأدخال عناصر مصفوفة أحادية من لوحة المفاتيح وعرضها؟

```
#include <iostream>
Void main ()
{  const int size =3;
Int Matrix [size];
Cout<<" enter matrix ";
For (int i=0 ; i<size ;i++ )
{ Cout<<i<< ":";
Cin>> matrix [i];
}
Cout<<" array element save ";
For ( int i=0 ; i<size ;i++ )
{   Cout<< matrix [i] <<"\n";}
}
```

4.5) تمهيد عناصر المصفوفة Initializing The Elements of Array

يمكن تخصيص عناصر المصفوفة بقيمها مباشرة ضمن قوسين معقوفتين والشكل العام لها:

arrayType arrayName[i]= { N_0 , N_1,.....n_{i-1} };

حيث أن:

arrayType – نوع المصفوفة arrayName

(index) i – دليل المصفوفة وهو اختياري (يمكن إهماله) ويمثل عدد عناصر المصفوفة.

N_0 – العنصر ذي الدليل الصفري هواول عناصر المصفوفة.

N_{i-1} – العنصر الأخير من عناصر المصفوفة الذي ترتيبة n–1 وتُستخدم الفاصلة لفصل عناصر المصفوفة عن بعضها.

{ } الاقوس المعقوفة– تُستخدم لتحديد قائمة بقيم عناصر المصفوفة وقد تكون أسماء مصفوفات فرعية أخرى.

إذا تم تحديد قيم لدليل المصفوفة أكثر من العدد المحجوز لها سيعترض المترجم على ذلك ويعيد رسالة يخبر فيها عن وجود خطأ، اما إذا كان دليل المصفوفة أقل من عدد عناصر المصفوفة المحجوزة لها مكان في الذاكرة سيملأ المترجم بقية العناصر (مواقعها في الذاكرة) بأصفار(0 للاعداد الصحيحة و0.0 للاعداد الحقيقية أو فراغات إذا كانت المصفوفة من النوع المحرفي char، أو سلسلة المحارف char string الخ وذلك تبعاً لنوعها) وإذا لم تمهد عناصر المصفوفة غير المستخدمة بالقيمة صفر قد يعطي المترجم قيما عشوائية لهذه الأماكن من الذاكرة، لذا يجب الانتباه وتصفير العناصر الغير مستخدمه من المصفوفة. فللتصريح عن مصفوفة عددية Age[ ] ذات بعد واحد وتتسع لأربعة أعداد صحيحة ولتكن (10 , 56 , 45 , 33) نكتب العبارة الآتية:

```
int Age[4]= { 33 , 45 , 56 , 10 };
```

مثال 2:

ليكن لدينا مصفوفة أحادية من 10 عناصر مدخلة من لوحة المفاتيح والمطلوب :

1. **إدخال عناصر المصفوفة.**
2. **طباعة عناصر المصفوفة.**
3. **حساب المتوسط الحسابي لعناصر المصفوفة.**
4. **أيجاد العنصر الأكبر من عناصر المصفوفة.**

```
1. include <iostream>
2. int main ()
3. {  Int X [10];
4. Int sum=0;
5. Float average;
6. Int Max;
7. Cout<<" enter matrix ";
8. For (int i=0 ; i<10 ;i++ )
9. {
10.Cin>> X [i];
11.}
12.Cout<<" array element save ";
13.For ( int i=0 ; i<10 ;i++ )
14.{    Cout<< X [i] <<"\n";
15.}
16.For (int i=0 ; i<10 ;i++ )
17.{
18.sum+= X[i]; }
19.average = sum/10;
20.Max = X[0];
21.For (int i=0 ; i<10 ;i++ )
22.{ if ( x[i] > Max )
23.Max= X[i]; }
24.Cout << "average is"<<average<<"\n";
25.Cout <<"max is "<< Max; }
```

ملاحظة:

عند إيجاد العنصر الأكبر من عناصر المصفوفة نبدأ المقارنة من العنصر الأول لذلك نفرض قيمة Max هي قيمة العنصر الأول من عناصر المصفوفة أي[0]Max= X

(مثال3): اكتب برنامجا يسمح بتنفيذ الآتي:

- إدخال قيم عناصر مصفوفة الأعداد الحقيقية مضاعفة الدقة a وعدد عناصرها(حجمها) 4.
- إظهار قيم عناصر المصفوفة a حسب دليلها i وبالترتيب العكسي.

_ ترتيب عناصر المصفوفة تصاعدياً حسب خوارزمية التعويم

_ طباعة عناصر المصفوفة بعد الترتيب

```
26.include <iostream>
27.const int size =4 ;
28.int main ()
29.{ double a [size];
30.Int n;
31.Cin>> n;
32.Cout<<" enter matrix ";
33.For (int i=0 ; i<n ;i++ )
34.{
35.Cin>> a [i];
36.}
37.Cout<<" array element save ";
38.For ( int i=n ; i>0 ;i-- )
39.{    Cout<< a [i] <<"\n"; }
40. Int L;
41.For (int i=0 ; i< n-1 ;i++ )
42.{ For (int j=0 ; j< n- i-1 ;i++ )
43.{ if ( a[j] > a[j+1] )
44.{ L= a[j];
45.a[j]= a[j+1];
46.a[j+1]= L; }
47.}}
48.For ( int i=0 ; i<n ;i++ )
49.{    Cout<< a [i] <<"\n";
50.}
51.}
```

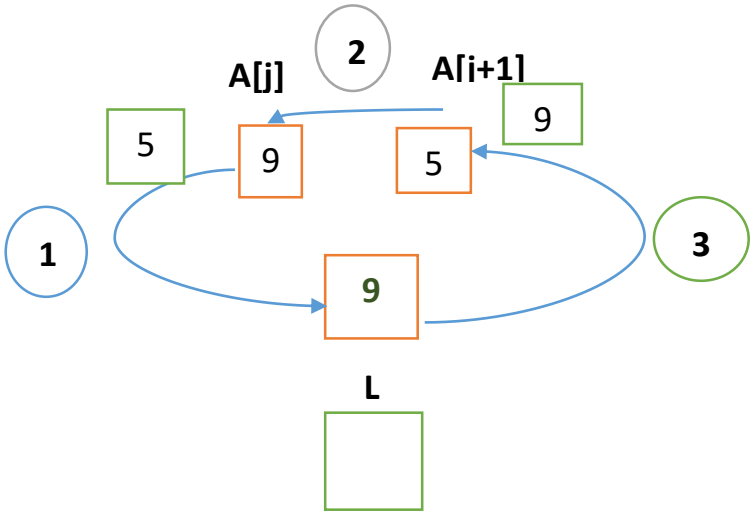
الحد الأعظمي لعدد عناصر

العدد الفعلي لعناصر المصفوفة

حلقة لعدد مرات التعويم

حلقة لدليل العنصر

مثال لتوضيح عملية التبديل



وظيفة:

اكتب برنامج يتعامل مع مصفوفة أحادية من الأعداد الصحيحة عدد عناصرها بحجم لا يتجاوز 20 عنصر. المطلوب:

1. **إدخال وطباعة عناصر المصفوفة.**
2. **طباعة عناصر المصفوفة مضاعفة.**
3. **استبدال العناصر ذات الترتيب الفردي بالرقم 1.**