

Computer Science 2

مراجعة تحضيرية لامتحان التقويمي الثاني

مدخل الى المصفوفة ذات البعد الواحد:

نعلم أنه في البرمجة أي متغير يتم الإعلان عنه في البرنامج يجب أن يخصص له مكان في الذاكرة لتخزينه. تختلف سعة المكان (عدد البايتات اللازمة لتخزين المتغير) حسب نوع المتغير... float, double, long . الآن السؤال المطروح هو إذا احتجنا الى متغيرات كثيرة في البرنامج مثلا من أجل معالجة بعض البيانات أو تضع نقاط الطلبة في هذه الحالة لا يمكنك ان تقوم بالاعلان العادي

float note1 ;

float note2 ;.....

بل في هذه الحالة نحتاج الى مصفوفة هي التي تختصر علينا هذا العمل كالاتي

float note[5] ;

هنا float هي طبيعة الأعداد ولأن نقاط الطلبة ستكون بالفاصلة.

note هي اسم المصفوفة.

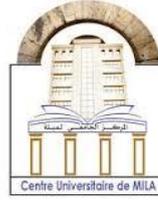
العدد 5 بين عارضتين يمثل عدد العناصر في المصفوفة أي أننا سندرج خمسة نقاط، والخمسة تعني سعة او حجم او طول المصفوفة (عدد العناصر) length أو size بالانجليزية.

لنحاول مع بعض تبسيط الأمر بمثال.

مثال 1:

الآن لدينا هذا الجدول الذي به خمسة علامات لخمس مواد

| المادة | رياضيات | انجليزية | اعلام آلي | فيزياء | تاريخ العلوم |
|----------------|---------|----------|-----------|--------|--------------|
| العلامة (note) | 9.75 | 10 | 7.5 | 12 | 15 |



Computer Science 2

الآن وباستعمال المصفوفات في البرمجة سنرى أنه من السهل ترتيب هذه العلامات في مصفوفة ذات بعد واحد

```
float note[5]={9.75, 10, 7.5, 12, 15};
```

الآن السؤال هو كيف سيقوم البرنامج C بترتيب هذه القيم في الذاكرة :

أولا يجب أن نتفق أن كل علامة في هذه المصفوفة أعلاه تمثل عنصرا في المصفوفة وكل عنصر هو بمثابة متغير وكل متغير سيحجز مكانا في الذاكرة. الآن تعال معي لنرى كيف تترتب هذه العلامات او العناصر في الذاكرة

| note[4] | note[3] | note[2] | note[1] | note[0] |
|---------|---------|---------|---------|---------|
| 15 | 12 | 7.5 | 10 | 9.75 |

تلاحظ معي أنه تم حجز خمس أماكن في الذاكرة لخمس متغيرات كل متغير يسمى باسم المصفوفة المعلن عنها (note) متبوعة بالأندكس index أي رتبة المتغير او العنصر في المصفوفة او الذاكرة . يجب ان نلاحظ ان رتبة المتغير تبدأ من الصفر 0 . اي ان العنصر الأول ستكون رتبته 0 والعنصر الأخير رتبته طول المصفوفة ناقص واحد اي 4=1-5 اي في هذا المثال العنصر الاخير رتبته اربعة.

اي أنه بدلا من انشاء 5 متغيرات والاعلان عنها هو سيقوم تلقائيا بذلك وسيسهل التعامل معها وهنا تكمن فائدة المصفوفات.

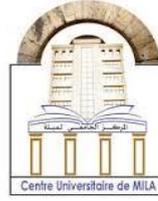
الآن اذا أردت أن أطبع العلامة الأولى يكفي فقط أن أكتب البيان التالي

```
printf("%f", note[0]);
```

وسيعرض لنا القيمة الأولى والتي هي علامة الرياضيات اي 9.75.

إذا أردت أن أطبع كل العناصر هنا انتبه معي سنستعمل حلقة for كالتالي:

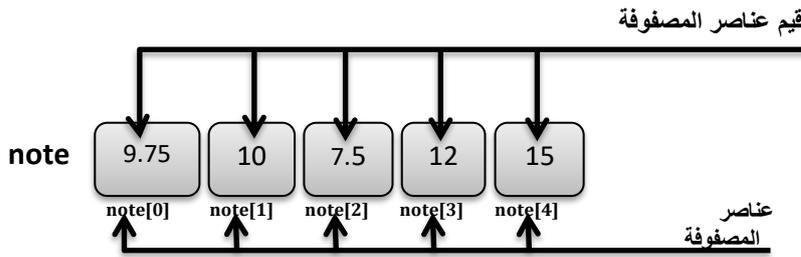
```
for(i=0 ; i<5 ; i++)
```



Computer Science 2

```
{ printf("%f\n", note[i]) ;}
```

لاحظ معي أن i هنا يمثل رتبة العنصر يبدأ من الصفر وينتهي عند اربعة لذا نكتب $i < 5$ وليس $i \leq 5$.



مثال 2:

في هذا المثال سنوضح بعض الاستعمالات للمصفوفات ذات البعد الواحد، بحيث سنعطي برنامجاً يوضح لنا التالي: معاينة نقاط بعض المواد ومعاملاتها والمجموع والمعدل النهائي للطالب كما يوضح الجدول التالي:

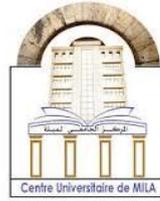
| المادة | رياضيات | انجليزية | اعلام آلي | فيزياء | تاريخ العلوم |
|---|---------|----------|-----------|--------|--------------|
| العلامة (note) | 9.75 | 10 | 7.5 | 12 | 15 |
| المعامل (coef) | 5 | 2 | 2 | 4 | 1 |
| العلامة \times المعامل | | | | | |
| مجموع النقاط | | | | | |
| مجموع المعاملات | | | | | |
| المعدل = مجموع النقاط / مجموع المعاملات | | | | | |



Computer Science 2

خطوات إنجاز هذا البرنامج:

- 1- لدينا ستة خانات في هذا الجدول إذن سيتم الإعلان عن ست متغيرات متغير لتخزين العلامة ومتغير للمعامل ومتغير يخزن قيمة العلامة في المعامل و متغير لمجموع النقاط ومتغير لمجموع المعاملات وآخر للمعدل.
- 2- العلامة بها خمس قيم إذن سيتم الإعلان عنها كمصفوفة ذات بعد واحد وسيتم ارفاق العلامات للمواد كقيم لعناصر هذه المصفوفة.
- 3- الإعلان عن مصفوفة ذات بعد مع ارفاق قيم معاملات المواد لعناصر هذه المصفوفة.
- 4- الإعلان فقط عن مصفوفة ذات بعد من أجل تخزين علامات المواد مضروبة في المعامل.
- 5- الإعلان عن متغير لتخزين قيمة مجموع النقاط (هنا المتغير به قيمة واحدة لذا لا يتم الاعلان عنه كمصفوفة)
- 6- الإعلان عن متغير لتخزين قيمة مجموع المعاملات
- 7- الإعلان عن متغير لتخزين قيمة المعدل.
- 8- نحتاج متغيرين آخرين لحلقتي for واحدة لارفاق قيم المصفوفة التي بها علامات المواد مضروبة في المعامل وأخرى لحساب مجموع المعاملات ومجموع النقاط.
- 9- في الأخير نقسم مجموع النقاط على مجموع المعاملات لنجد معدل هذا الطالب.
- 10- إذا تحصل الطالب على معدل أكبر من عشرة فهو ناجح وعذا ذلك يعتبر راسب.



Computer Science 2

بعد كل ذلك نتحصل على البرنامج التالي:

```
#include <stdio.h>

int main() {
    float note[5]={9.75, 10, 7.5, 12, 15};
    float coef[5]={5, 2, 2, 4, 1};
    float A[5];
    float sumcoef=0, sumnote=0, moyen;
    int i,j;
    for(i=0; i<5; i++)
    {
        A[i]=note[i]*coef[i];
        printf("A[%d]=%f\t",i,A[i]);
    }
    printf("\n");
    for(j=0;j<5;j++)
    {
        sumcoef=sumcoef+coef[j];
        sumnote=sumnote+A[j];
    }
    printf("sumcoef=%f\nsumnote=%f\n",sumcoef,sumnote);
    moyen=sumnote/sumcoef;
    printf("moyen=%f\n",moyen);
    if(moyen>10)
    {
        printf("Admitted");
    }
    else
    {
        printf("Refused");
    }
    return 0;
}
```

Computer Science 2

في الأخير أطلب منكم الجدية في العمل والاجتهاد أكثر والتركيز على النجاح، وأتمنى النجاح لكم جميعا كلكم تستطيعون ذلك، إذ أنكم كلكم تتمتعون بالذكاء الخارق أنتم فقط لاتدركون الطريق إلى ذلك ولا توظفون ذلك، كما أرجو أن تبتعدوا عن الغش.

