بسم الله الرحمن الرحيم

Principles Of Programming Languages

مبادئ لغات البرمجة

عدد الساعات: ٢ نظري+٢ عملي

الرمز: 211 حسب

المتطلبات: 111 حسب (مقدمة في الحاسب)

أستاذات/المادة: م. نجلاء حسن

Lecture 3:

- انواع البيانات Data Type
- انواع البيانات البسيطة في C++
- انواع البيانات المركبة في C++

أنواع البيانات Data Type

١- أنواع البيانات الأولية Primitive Data Type

كل لغات البرمجة تحتوي على مجموعة من أنواع البيانات الأولية، وتستخدم هذه الأنواع الأولية مع نوع أو أكثر من منشئ النوع لتكوين أنواع البيانات الهيكلية، وتنقسم إلى: رقمي integer ، وحقيقة floating point وحرفية character ومنطقية Boolean

1.1. أنواع البيانات الرقمية Numeric Types

- أكثر الأنواع شيوعاً هو نوع البيانات الرقمي Integer. تدعم لغات البرمجة أحجام مختلفة من نوع البيانات الرقمي مثلاً: short integer, long integer, unsigned short integer, unsigned long integer .
- يمثل قيمة نوع البيانات integer في الحاسوب بسلسلة من البت مع حفظ آخر بت في اليسار لتمثيل الاشارة (+أو-) .

2.1. أنواع البيانات الحقيقية Floating point Data Type

- تمثل الأعداد الصحيحة تقريب لأغلب الأعداد الحقيقية.
- من المشاكل التي تظهر عند استخدام تمثيل الاعداد الحقيقية هو عدم الدقة في العمليات الحسابية .
- تخزن في اربع وحدات بايت من الذاكرة.

3.1. الأعداد الكسرية Decimal

تستخدم الأعداد Decimal لتخزين عدد أرقام عشرية ثابتة، مع الشرطة **العشرية** في مكان ثابت في القيمة، وتستخدم عادة في مجال الأعمال التجارية.

تخزن الأعداد كما في السلاسل الحرفية باستخدام الشفرة الثنائية للأرقام فيما يسمى بالتمثيل المشفر وتحتاج إلى مساحة تخزين أكبر من التمثيل الثنائي.

4.1. نوع البيانات الحرفية Character Type

تخزن بيانات الحروف في الحاسوب في شكل شفرات أرقام أشهر طريقة للشفرة هي شفرة ASCII (American Standard Code for Information Interchange)

5.1. نوع البيانات منطقي Boolean Type

ان نوع البيانات منطقي هو أبسط أنواع البيانات لأنها تحتوي على قيمتين فقط هما عنصر للخطأ وعنصر آخر للصواب. حيث أن كل القيم غير الصفرية تعتبر صواباً (True) والصفر تعتبر خطأ (false) وأيضاً تستخدم نوع البيانات منطقي لتمثيل التبديل في البرامج يمثل النوع منطقي بوحدة بت مفردة .

6.1. نوع البيانات السلاسل الحرفية Character String Type

نوع السلاسل الحرفية هي التي تتكون من قيمة متتالية من الحروف، وتستخدم ثوابت السلاسل لتمثيل الإدخال والإخراج وكل أنواع البيانات . ويستفاد منها في عمليات معالجة الحروف .

2. أنواع البيانات المعرفة Defined Ordinal Type

هي أنواع بيانات يعرفها المستخدم بالطريقة التي يريد ويستطيع تغييرها والتحكم بمساحتها.

3. أنواع البيانات مصفوفة Array Type

تعريف:

المصفوفة : عبارة عن مجموعة من العناصر من نوع واحد (مثلاً integer)
من حيث إن كل عنصر يحدد بموقعه داخل المجموعة مقارنة بالعنصر الأول .

1.3. ملاحظات حول المصفوفات

- أغلب برامج الكمبيوتر تحتاج إلى برمجة مجموعات من البيانات من نفس النوع وتجري عليها نفس العمليات، لذا تستخدم المصفوفات.
- يمكن أن تكون المصفوفة من بعد واحد أو متعددة الأبعاد .
- الصيغة المستخدمة لتمثيل المصفوفات تقريباً واحدة في كل لغات الكمبيوتر ألا وهو اسم المصفوفة متبوعاً بالفهرس داخل قوس مربع أو قوس دائري في بعض اللغات مثل Fortran و Ada .

4. نوع البيانات سجل Record Type

نوع البيانات سجل هو عبارة عن مجموعة عناصر مختلفة من البيانات، حيث أن كل عنصر يعرف بإسم مختلف ويسمى الحقل .

مثال لذلك :

لتمثيل المعلومات حول طالب جامعي يتطلب ذلك وجود اسم الطالب ، رقم الطالب، مجموعة الطالب .... وهكذا.

حيث يمكن استخدام نوع البيانات حرف لتمثيل اسم الطالب ونوع البيانات رقم لتمثيل رقم الطالب والمجموع وهكذا.

1.4. ملحوظات حول السجلات

1- توجد السجلات في كل لغات البرمجة ما عدا النسخة الأولية من Fortran 90

2 - في لغات البرمجة غرضية التوجه نجد أن الفئات تدعم السجلات.

3 - تختلف السجلات عن المصفوفات في أنها ليس من الضروري أن تحتوي على بيانات من نوع واحد. بالإضافة إلى أنها لا تمثل بواسطة الفهرس.

4 تستخدم السجلات في حالة أن تكون هنالك مجموعة من البيانات من نوع مختلف.

م. نجلاء حسن

5. نوع المؤشرات Pointers Type

نوع المؤشرات عبارة عن متغير يحمل قيماً لعناوين في الذاكرة مع قيمة خاصة تسمى : nil

تستخدم المؤشرات للأغراض التالية:

- تستخدم في العنونة غير المباشرة للمتغيرات كما في لغة أسمبلي
- تستخدم في إدارة الذاكرة الديناميكية .

أنواع البيانات الأساسية في لغة ++ c

اسم النوع	يستعمل لتخزين	أمثلة عن القيم المخزنة
char	أحرف	''a''
short	أرقام صحيحة قصيرة	222
int	أرقام صحيحة عادية الحجم	153,406
long	أرقام صحيحة طويلة	123,456,789
float	أرقام حقيقية قصيرة	3,7
double	أرقام حقيقية مزدوجة	7,533,039,395
long double	أرقام حقيقية ضخمة	9,176,321,236,01202,6

١- الاحرف char

يتم تخزين الأحرف في متغيرات من النوع char

```
char ch;
```

تنشئ مساحة من الذاكرة لحرف وتسميه ch. لتخزين حرف ما في هذا المتغير نكتب

```
ch='z'
```

ودائماً تكون الأحرف الثابتة كـــ 'a' و'b' محصورة بعلامة اقتباس فردية.

يمكن استعمال المتغيرات من النوع char لتخزين أرقام كاملة بدلاً من أحرف ، فمثلاً يمكننا كتابة:-

```
ch=2;
```

نطاق القيم الرقمية التي يمكن تخزينها في النوع char يتراوح بين 128- إلى 127

٢- الاعداد الصحيحة integer

هنالك ثلاثة أنواع من الأعداد الصحيحة في C++

١- قصير short
٢- عدد صحيح int
٣- طويل long

الجدول التالي يبين هذه الانواع و المساحة التي تأخذها من الذاكرة و نطاق الارقام التي يمكن ان تأخذها.

اسم النوع	الحجم	النطاق
char	1byte	-128 إلى 127
short	2byte	-32,768 إلى 32,767
int	مثل short في أنظمة 16bit ومثل long في أنظمة 32bit	
long	4byte	-2,147,483,648 إلى 2,147,483,647

٣- الاعداد الصحيحة الغير معلمة (بدون اشارة) Unsigned

- اذا حذفنا الاشارة المصاحبة لأنواع البيانات الصحيحة والحرفية، يتم تغيير مدى القيم التي تستطيع هذه الأنواع تخزينها. لأننا نتعامل مع القيم الموجبة فقط

اسم النوع	الحجم	النطاق
unsigned char	1byte	0 إلى 255
unsigned short	2byte	0 إلى 65,535
unsigned int	مثل unsigned short في أنظمة 16bit ومثل unsigned long في أنظمة 32bit	
unsigned long	4byte	0 إلى 4,294.967.295

٤- الارقام العائمة (العشرية) float

يتم استعمال الأرقام العائمة لتمثيل قيم يمكن قياسها كالأطوال أو الأوزان.

هنالك ثلاثة أنواع من الأرقام العائمة في أنظمة التشغيل الشائعة الاستعمال

اسم النوع	الحجم
float	4byte
double	8byte
long double	10byte

البيانات المركبة في C++

١. البيان struct

- ✓ يتكون من عدد من المتغيرات/البيانات البسيطة (عددية ، حرفية ، ...) + دوال (وظائف) تعمل علي هذه البيانات.
- ✓ في الغالب لا نتعامل مع التركيبة بل نتعامل مع كائن object مشتق منها.

الصيغة العامة للاعلان عن تركيبة

```
struct struct_name
{
    member variables
    member functions
} object_name ;
```

اسم التركيبة

متغيرات التركيبة

دوال التركيبة

بداية و نهاية التركيبة

الكائن المشتق

طرق اشتقاق كائن من التركيبة

١. بجوار نهاية التركيبة "}" قبل ";" .
٢. في جملة مستقلة عن التركيبة في أي مكان بالبرنامج كما يلي:

```
struct student ss , tt ;
```

مثال:

انشاء (الاعلان عن) تركيبة لطالب student تحتوي علي الرقم الجامعي ، المعدل التراكمي ، اسم الطالب ، ثم اشتقي منها الكائن s لتخزين بيانات فيه.

```
struct student
{
    long number;
    float average;
    char name[30];
}s;
```

التعامل مع الكائن s:

```
s.number=2211240;
s.average=4.3;
strcpy(s.name,“mohamed”);
```

الاعلان عن نوع بيانات من تركيبة struct

```
typedef struct
{
    long number;
    float average;
    char name[30];
} student;
```

نستطيع الآن الاعلان عن أي متغير من نوع student دون استخدام كلمة struct

```
Student s1,s2;
```

٢. البيان متعدد الكلمات enumerator (enum)

- ✓ يسمح بالتعامل مع مجموعة كلمات كل واحدة لها رقم بحسب ترتيبها.
- ✓ الكلمة الاولي تأخذ الرقم صفر ، الثانية الرقم ١ ، ... .

مثال :

للإعلان عن متعدد الكلمات (ايام الاسبوع)

```
enum day{sat , sun , man , tus , wed , thr , fri};
```

Sat يأخذ الرقم صفر ، ...

مثال :

```
enum student {primary , prep , secondary , unv};
int n=prep;
cout<<n;      // سيطبع القيمة ١
```

٢. البيان طبقة Class

- ✓ هي اساس احد العلوم الهامة (علم البرمجة الموجهة الاهداف object oriented programming) . هذا العلم مبني علي ان اساس ان كل شيء أو كائن في الحياة يمكن ان يتبع صنف او فصيلة أو طبقة معينه.
- ✓ هذا الشيء يسمي بالكائن object ينتمي الي طبقة class و كل طبقة تنحدر من طبقة اعلي.
- ✓ تحتوي الطبقة علي عناصر البيانات + الدوال (الطرق methods) التي تعمل علي تلك البيانات (معالجة بيانات الطبقة).
- ✓ من الممكن ان تحتوي الطبقة علي بيانات بدون دوال.
- ✓ عناصر البيانات و الدوال تسمي بأعضاء الطبقة (member data , member function).
- ✓ نستطيع اشتقاق أي عدد من الكائنات من الطبقة للعمل عليه داخل الدالة الرئيسية.
- ✓ موقع الطبقة قبل الدالة الرئيسية.

مثال :

انشئ طبقة لموظف employee تحتوي علي بيانات الموظف (الاسم، الرقم الوظيفي، الراتب.

```
class employee
{
    public:          // بيانات عامة
        int emp_no;
        float emp_sal;
        char emp_name[30];
};
void main( )
{
    class employee e1;
    employee e2,e3;
    cin>>e1.emp_no>>e1.emp_sal>>e1.emp_name;
    ...
}
```

اسم الطبقة

بداية الطبقة

بيانات الطبقة

نهاية الطبقة

اشتقاق كائنات

ملئ بيانات الكائن