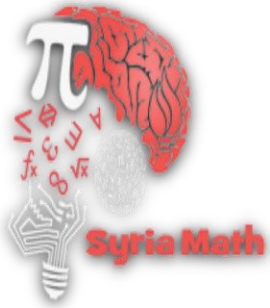


◄ دكتور المادة: سمير جعفر

◄المحاضرة :الثامنة عنوان المحاضرة :الصفوف

المحتوى العلمي : أهلاً بكم أصدقائي في المحاضرة الثامنة من مقررنا الخوارزميات والبرمجة(2)

سنتناول أول مفهوم من مفاهيم البرمجة الغرضية التوجه وهو الصفوف.

ظهرت البرمجة الغرضية التوجه اختصاراً(oop) تلبية لحاجة المبرمجين للقدرة على التعامل مع البرامج متزايدة التعقيد .

الصفوف في لغة C++(Classes):

تعد الصفوف من أهم ميزات C++ فهي الألية التي تستخدم لإنشاء الأغراض(objects).

الصف :هو أداة برمجية تمكننا من إنشاء نوع جديد بالإضافة الى تعريف العمليات الممكنة على هذا النوع بالإضافة الى تقييد الوصول الى محتويات هذا النوع .

التصريح عن صف يكون بالشكل التالي:

class اسم الصف{

محدد وصول

معطيات أعضاء

دوال أعضاء(طرق)

};

المعطيات الأعضاء:عبارة عن حقول(متحولات من أنواع معرفة مسبقا بسيطة او مركبة او من نوع سجل ....).

الدوال الأعضاء:دوال عادية تقوم باجراء عمليات على المعطيات الأعضاء ويمكن تسمية الدوال ضمن الصف باسم طرق(methods).

❖ يستطيع المبرمج تحديد من يمكنه الوصول إلى هذا الصف و استخدامه و يحدد ذلك أثناء كتابة الصف عن طريق محددات الوصول الثلاثة (public,private,protected) في المحاضرة تناولنا **محدد الوصول العام (public)** :فللتصريح عن الأعضاء العامة في الصف نستخدم الكلمة المحجوزة public متبوعة بنقطتين (:) ويمكن الوصول الى الأعضاء العامة بعد كلمة public من قبل أي جزء اخر من البرنامج الذي يحتوي الصف .

❖ بدأنا التصريح عن الصف بالكلمة المفتاحية (class) وهي كلمة محجوزة في لغة C++ (تكتب بأحرف صغيرة).

❖ عند نهاية الصف نغلق القوس { } ونضع بعده فاصلة منقوطة.

اتفق المبرمجون على كتابة اسم الصف بأن يبدأ بحرف كبير (ليست قاعدة أي عدم الالتزام بالاصطلاح السابق لا يؤدي إلى خطأ في البرنامج)

➢ أصبح لدينا نوع جديد يدعى Class حتى نستخدم هذا النوع الجديد يجب التصريح عن متحولات منه

التصريح عن متحول من نوع صف(يسمى المتحول من نوع الصف ب غرض : object):

اسم النوع (الصف) اسم المتحول (الغرض) ;

مثال:

```
Class Myclass {
public:
int x;
double y;
double mult ( ){
return(x*y);}
};
```

محدد الوصول عام أي كل الأعضاء التي تاتي بعده يمكن الوصول لها من داخل او خارج الصف.

المعطيات الأعضاء:عنصرين احداهما متحول من نوع صحيح والثاني متحول من نوع حقيقي

دالة لحساب ضرب العددين

لاتنسى الفاصلة المنقوطة في نهاية الصف

للتصريح عن متحول من نوع الصف السابق يمكن أن نكتب:

Myclass mc ;

عندما نكتب $int\ x\ ;$ نكون قد صرحنا عن متغير من نوع صحيح وبنفس الوقت فإنه يحجز مكان (صندوق) في الذاكرة ويعطيه اسم x .

عندما نكتب

$$struct\ complex\ \{double\ x;\ double\ y;\}$$

$$complex\ c;$$

نكون قد صرحنا عن متحول من نوع السجل ويحجز في الذاكرة صندوق c يحوي صندوقين x,y إذن تعريف متحول يقتضي حجز في الذاكرة وهذا الكلام صحيح الا في حالة الصفوف فالتعريف ليس له علاقة بالحجزفقط يعلم الكومبايلر بأنه يوجد صف وعملية الحجز هنا هي عملية بناء للغرض واعطاؤه قيمة ابتدائية يتم بناء الغرض (متحول من نوع صف) عن طريق **الباني** وهو عبارة عن طريقة(دالة داخل الصف)مميزة لها اسم الصف نفسه وليس لها نوع ارجاع .

- الباني يقوم بحجز مكان في الذاكرة واعطاؤه قيمة ابتدائية.يمكن ان يحتوي الصف على عدة بواني (الباني له نفس اسم الصف لذلك بناء عدة بواني يقتضي ان تحمل هذه البواني نفس الاسم لذلك نعود لاستخدام مفهوم التحميل الزائد للدوال)
- وجود نوع دون بان ليس له معنى .الكومبايلر عندما يقرا الصف اذا انتها ولم يجد بان فاللغة سوف تولد باني هذا الباني هو الباني الافتراضي .
- الباني الافتراضي ليس له وسطاء(يعطي القيم الافتراضية للانواع).
- اذا وجد باني واحد على الأقل في الصف فاللغة لن تولد باني افتراضي .
- الباني الخالي هو الباني الذي يكتبه المبرمج في كود البرنامج اما الباني الافتراضي الذي تولده اللغة فلا نجده في الكود بل اللغة تضعه ضمن الملف التنفيذي للبرنامج .
- الوصول الى متحولات الأغراض من الصف نستخدم معامل النقطة (.).

مثال (اين الخطأ عند تنفيذ البرنامج):

```
#include <ioestream .h>
class Myclass{
public:
int x;
double y;
Myclass (int a,double b){x=a;y=b;}
```

Syria Math Team

```
Myclass (int a){x=a;y=0;}
Myclass(double b){x=0;y=b;}
double mult( ) {return(x*y);}
void print( ) {
cout<<"x="<<x<<endl;
cout<<"y="<<y<<endl;}
};
int main( ) {
Myclass mc1(3,2);
Myclass mc2(5);
Myclass mc3(2,1);
Myclass mc4( );
cout<<mc2.mult( ) ;
cout<<mc1.print( ) ;
return 0 ;}
```

هذا خاطئ :لايوجد في الصف باني خالي واللغة لن تولد باني افتراضي لوجود بواني أخرى في الصف

استدعاء دالة mult للغرض mc2

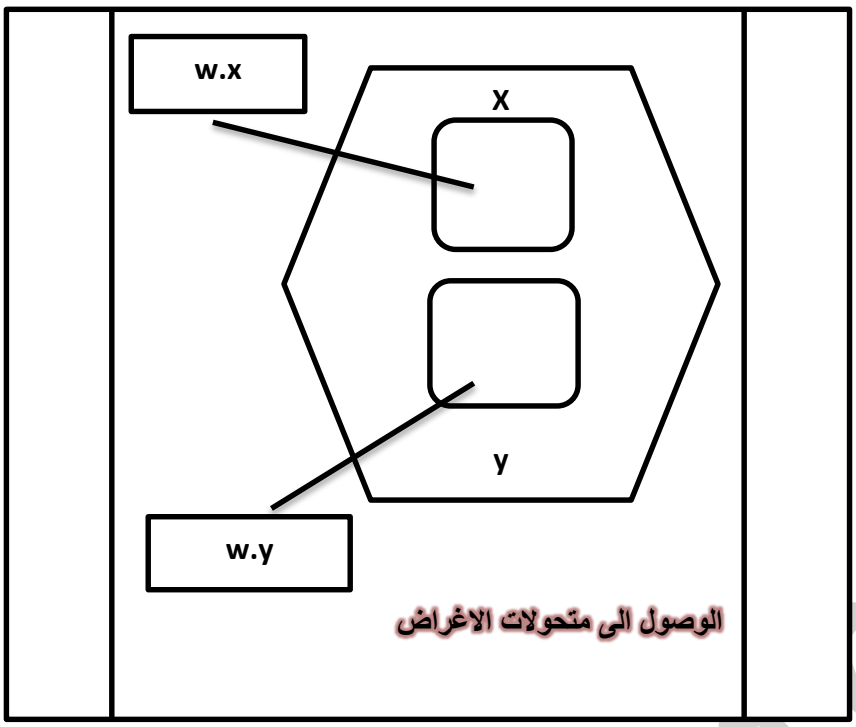
استدعاء دالة print للغرض mc1

☺---------

الباني الخالي :	الباني الافتراضي :
يقوم المبرمج بكتابته .	تقوم اللغة بإنشائه أثناء التنفيذ في حال عدم وجود أي باني داخل الصف .
لا يحوي وسطاء و يقوم بإعطاء القيم الافتراضية للمعطيات الأعضاء.	لا يحوي وسطاء و يقوم باعطاء القيم الافتراضية للمعطيات الأعضاء.
موجود في الكود البرمجي للبرنامج	موجود في الملف التنفيذي للبرنامج فقط

Syria Math Team

لاننسى



ان الصف ينتهي بفاصلة منقوطة

اسم الصف يبدأ بحرف كبير

بعد محدد الوصول لاننسا النقطتين(:)
ومحددات الوصول من الكلمات المحجوزة تكتب باحرف صغيرة.

Syria Math Team

انتهت المحاضرة

إعداد: عبير خزنة كاتبي &سندس درويش&وفاء شيخ سالم.