

كلية الاسراء الجامعة



قسم هندسة تقنيات الحاسوب

مادة البرمجة (C++)

المرحلة الاولى

2018-2017

المحاضرة الاولى

استاذة المادة: م.م غيداء احمد علي

- الهيكلية العامة للبرنامج
- المتغيرات
- دوال الإخراج
- دوال الإدخال
- العمليات الحسابية وطرق تمثيلها وأولوياتها
- تمثيل العمليات الرياضية
- السبقيات وطرق معاملتها
- إضافة تعليقات
- مؤثرات الزيادة والنقصان

الهيكلية العامة للبرنامج:

للبرنامج في لغة ++C شكل عام عند كتابته وهو ثابت تقريبا في أجزائه الرئيسية في كل البرامج وتكون طريقة كتابته بشكل التالي:

C++	الكود بلغة
<p>منطقة التعريفات العامة واستدعاء مكاتب للبرنامج</p> <pre># include<iostream.h></pre> <p>الدالة التي تكتب بداخلها اكواد البرنامج</p> <pre>Main()</pre> <pre>{</pre> <p>ساحة الاكواد أو منطقة كتابة اكواد البرنامج والقراءة والطباعة</p> <pre>}</pre>	

- منطقة التعريفات العامة واستدعاء مكاتب للبرنامج: في هذا المكان يتم كتابة جميع المكاتب التي سنحتاج إليها داخل البرنامج وكذلك المتغيرات التي تعرف بشكل عام لكل البرنامج والسجلات والدوال.
- دالة main: اول دالة تنفذ عندما يعمل البرنامج لذلك بداية البرنامج تكون من هذه الدالة
- ساحة كتبة الاكواد: هي المنطقة التي يتم بداخلها كتابة الاكواد البرمجية والتعريفات وغيرها
- يجب وضع فارزة منقوطة في نهاية أي تعبير مبرمج من قبل المستخدم تدل على أن التعبير انتهى.

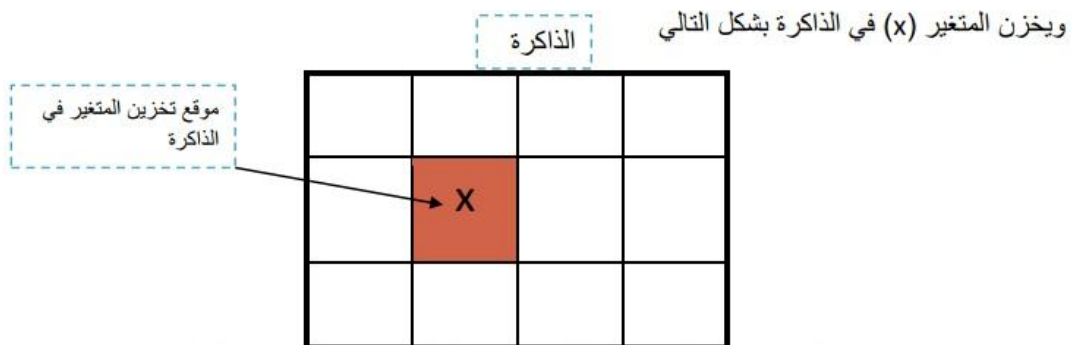
المتغيرات:

هي مواقع في الذاكرة تخزن فيها قيم معينة قد تتغير أثناء البرنامج أو قد تبقى ثابتة. وأسماء المتغيرات تكون مفتوحة حسب الرأي الشخصي للمبرمج ممكن أن يسميها إي اسم لكن يجب أن لا يكون الاسم من الأسماء المحجوزة مثل (**while,for,if**) أو أي أسم آخر محجوز من قبل المترجم ورغم أن تسمية المتغيرات مفتوحة لكن يفضل أن تكون تسمية المتغير دالة عليه مثال لو كان المتغير يدل على الوقت فيفضل تسميته **time** حتى تكون أكوادك واضحة وحتى لا يكون برنامجك متشابك كمعكرونة السبكتي لا يعرف القارئ بدايته من نهايته. المتغير الذي يتم تعريفه داخل البرنامج يجب تحديد نوعه. هناك عدة أنواع من المتغيرات يمكن تعريف المتغيرات الجديدة بها وهي:

تعريف المتغير	استخدامه
Int var;	يستخدم لتعريف المتغير من نوع integer أي رقمي مثلا (int x=5)
Float var;	يستخدم لتعريف المتغير من نوع كسري مثلا (Float var=5.4)
Char var;	يستخدم لتعريف المتغير من نوع حرفي مثلا (Char var='a')
Double var;	يستخدم لتعريف المتغير من نوع Double أي حقيقي
Void var;	يستخدم لتعريف المتغير من نوع Void أي لا يرجع أي قيمة

فمثال لو كان المتغير (x) يحوي قيمة متغير رقمي بدون فارزة بعد الصفر فيجب تعريفه تحت الدالة main() بأنه متغير من نوع integer.

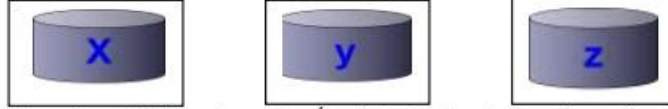
```
البرمجة بلغة C++
#include<iostream.h>
Main()
{
// هنا تعرف المتغيرات
int x;
}
```



- كل موقع في الذاكرة يكون مرقم برقم معين يختلف رقمه عن غيره من المواقع

✓ يمكن تعريف أكثر من متغير في سطر واحد بوضع فارزة بينهم

```
الكود
char x,y,z;
```



✗ المتغيرات (x,y,z) هي متغيرات تستطيع أن تخزن في داخلها حرف واحد فقط

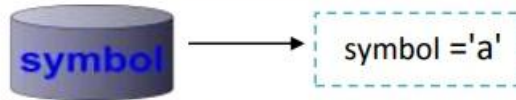
✓ يمكن أن يسند المتغير بقيمة مباشرة إثناء وقت التعريف هكذا

```
الكود
int x=5;
```



✓ قد يسند المتغير بقيمة معينة بعد التعريف في أي مكان في البرنامج .

```
الكود
char symbol;
symbol ='a';
```



✗ المتغير (symbol) تم تخزين الحرف a في داخله .

• الحروف عندما تخزن تضع بين علامة تنصيصية واحد دلالة على انها حرف واحد فقط

✓ قد نسند قيمة متغير إلى متغير آخر

```
الكود
int first ,second;
first =51;
second= first ;
```

أصبح المتغير second يحوي نفس قيمة المتغير first في الخطوة رقم 3 أو قد يسند المتغير بقيمة في وقت الادخال من لوحة المفاتيح سنتطرق عليه لاحقا.

اسناد قيم للمتغيرات:

تكون طريقة إسناد إي قيمة أو تعبير رياضي إلى متغير بشكل التالي

متغير أو تعبير رياضي = اسم المتغير

• في الطرف الايمن من المساواة يوجد فقط اسم المتغير الذي نريد إسناد قيم إليه

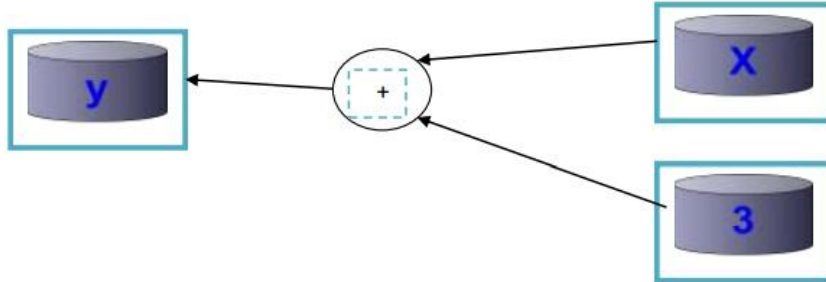
• في الطرف الايسر من المساواة نستطيع كتابة إي تعبير رياضي أو متغير أو قيمة معينة

مثال : إسناد قيمة تعبير رياضي إلى متغير (بطريقة صحيحة)

```
الكود
1.int y, x=5;
2.y=x+3;
```



• في الخطوة رقم (2) أسندنا القيمة الناتجة من تعبير رياضي وهو $(x+3)$ إلى المتغير (y)



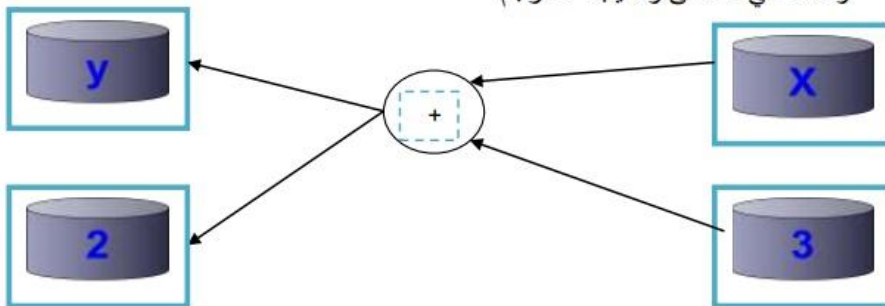
• لو تلاحظ أن الطرف الأيمن مكون من متغير فقط والطرف الأيسر مكون من تعبير رياضي

مثال : إسناد قيمة تعبير رياضي إلى متغير (بطريقة خاطئة)

```
الكود
1.int y, x=5;
2.y+2=x+3;
```



• في الخطوة رقم (2) أسندنا القيمة الناتجة من تعبير رياضي وهو $(x+3)$ إلى تعبير رياضي آخر وهذه الشيء خاطئ ولا يقبله المترجم



المتغيرات الثابتة: هي متغيرات تبقى قيمتها ثابتة أثناء تنفيذ البرنامج ولا تتغير أبدا وتعرف بوضع كلمة **const** قبل تعريف نوع المتغير. وفائدتها نستخدمها للمتغيرات التي لا نريد أن تتغير قيمتها أثناء عمل البرنامج أبدا. مثال قيمة **pi=3.14** هذه قيمة رياضية ولا تتغير ابد مهما حدث ودائما نحتاجها في البرامج الرياضية لذلك نعرفها **const**(ونعطيها قيمة 3.14)سنضمن لأنفسنا أنها لا تتغير مهما حدث وأيضا نحتاجها نكتب فقط **pi**