



كلية الهندسة المعلوماتية

مدخل إلى الخوارزميات والبرمجة

Introduction to Algorithms and Programming

أ. د. علي عمران سليمان

محاضرات الأسبوع الرابع

الفصل الثاني 2022-2023

expressions	<u>التعابير</u> 7-2-2	C++ مقدمة إلى لغة البرمجة
statements	<u>العبارات</u> 8-2-2	أجيال لغات البرمجة.
symbolic constants	<u>الثوابت الرمزية</u> 9-2-2	ـ فئة رموز لغة C++ .
Operators of C++	<u>معاملات لغة C++</u> 3-2	ـ المعرفات والكلمات المحفوظة
unary operators	<u>المعاملات الأحادية</u> 2-3-2	Identifiers and Key word.
Relational operators	<u>المعاملات العلائقية</u> 3-3-2	<u>ـ أنواع البيانات</u> 3-2-2
Logical operators	<u>المعاملات المنطقية</u> 4-3-2	<u>ـ الثوابت</u> 4-2-2
Assignment operators	<u>عوامل التخصيص أو النسب</u> 5-3-2	<u>ـ المتغيرات</u> 5-2-2
Conditional operator	<u>المعامل الشرطي</u> 6-3-2	<u>ـ المصفوفات</u> 6-2-2

المحاضرة من المراجع :

[1]- Deitel & Deitel, C++ How to Program, Pearson; 10th Edition (February 29, 2016)

[2]- د. علي سليمان, مدخل إلى الحاسوب والخوارزميات, جامعة تشرين 2005-2006

1-2 مقدمة إلى لغة البرمجة C++

- **اللغة:** هي نسق من الإشارات والرموز، تشكل أداة من أدوات المعرفة، وتعتبر اللغة أهم وسائل التفاهم والاحتكاك بين أفراد المجتمع في جميع ميادين الحياة. وبدونها يتذرع نشاط الناس المعرفي. ترتبط اللغة بالتفكير ارتباطاً وثيقاً؛ فأفكار الإنسان تصاغ دوماً في قالب لغوي، حتى في حال تفكيره الباطني.
- **لغة البرمجة:** هي بالأساس طريقة تسهل للمبرمج كتابة برنامجه في هيئة تعليمات وأوامر يفهمها الحاسوب بغرض تنفيذ العمل المطلوب.
- **لغة البرمجة C++:** هي لغة برمجة عالية ومنخفضة المستوى، كائنة التوجّه، متعددة الأ Formats، محمولة، متعددة المهام، شاملة، قوية، سريعة إذ تستعمل في العديد من أنظمة التشغيل، اقتصادية، وسهلة الفهم وقابلة للتعديل والتصحيح بسهولة مما يوفر الجهد والمال.



أجيال لغات البرمجة

الجيل الأول لغة الآلة : Machine Language تتألف من مجموعة من الأصفار والواحدات التي تعطي الأوامر إلى الحاسب من أجل تنفيذ تعليماته كل تعليمة على حده وبالتالي تصبح صعبة على المبرمجين، **لاتطلب** مقتضيات أو مترجمات والباقي يتطلب.

الجيل الثاني لغة التجميع : جمعت بحيث تتضمن التعليمية الواحدة أكثر من تعليمية بلغة الآلة.

الجيل الثالث اللغات الإجرائية procedural languages : هي نمط برمجة مشتقة من البرمجة المهيكلة تستند إلى مفهوم إستدعاء الإجراء، المتضمنه رموز رياضية ومنطقية معروفة وتنادى لمعطيات مختلفة وفي أوقات مختلفة.

الجيل الرابع اللغات عالية المستوى High-Level Language : تمتاز هذه اللغات بالبساطة من حيث الفهم والاستخدام وتتألف من مكتبات بأدوات جاهزة منحت المبرمج القدرة على تطوير البرامج وامتازت بالواجهات الرسومية.

الجيل الخامس اللغات الطبيعية Natural Language: وهي لغات برمجة منطقية مبنية على القيود وعليها الاستنتاجات دون حاجة المبرمج لمعرفة الدقيقة كيف يتم كتابة الكود بالتفصيل، تستخدم هذه اللغات بشكل رئيسي في برامج الذكاء الصناعي.

- **المترجم Compiler** : يقوم بترجمة البرامج الموجودة بلغة عالية المستوى أو لغة التجميع (برنامجه المصدر source program) إلى لغة الآلة (برنامجه الهدف object program) وبعدها يمكن تنفيذها، تحتاج المترجمات إلى زمن طويل في الترجمة وزمن قصير في التنفيذ.

- **المفسر Interpreter** : يفسر كل سطر خلال وبعد الانتقال إلى السطر الذي يليه ويحتاج إلى زمن قصير في التفسير وطويل في التنفيذ لذا تستخدم المفسرات عند تطوير البرامج وبعدها يتم ترجمتها.

1-2-2 - فئة رموز لغة C++



تتألف أبجدية لغة C++ من :

- المحارف الكبيرة والصغيرة A – Z (وتميز لغة C و C++ بين المحارف الصغيرة والكبيرة).
- الأرقام من 0 إلى 9.
- بعض الرموز الخاصة كعناصر لبناء هيكل البرنامج. ونسرد منها:

! * + \ “ < # (= | {) % ≈ ; } . ^ [! , ? & - } . (blank)

كما وتوجد رموز أخرى منها @ , \$، وتستخدم لغة C++ العديد من بدائل الخلط بين هذه الرموز مثل (\L,\n,\b) الحركة الخلفية لمسافة واحدة back space، السطر الجديد، والجدول الرئيسي على التوالي وهذا ما يعرف بـ تتابعات أو تسلسلات الهروب



2-2-2- المعرفات والكلمات الممحوزة Identifiers and Key words

المعرفات ما هي إلا أسماء عناصر البرنامج المختلفة، مثل المتغيرات والتوابع والمصفوفات. وقواعد تسمية المعرفات:

- مهارف وأرقام إلا أن ذلك يشترط أن يكون الرمز الأول في المعرفة حرف ويجوز أن يكون تسطيره سفلية أو رمز \$.
- أن لا يحوي على الرموز الخاصة.
- كذلك يجب أن لا يحتوي المعرف على فراغ.
- أن لا يكون كلمة ممحوزة.

لغة C تميز بعدد 8 مهارف فقط من الأسماء الطويلة،

لغة C++ بطول معرفات حسب الرغبة وقد تكون بعض أجياته 31 حرفا.

الكلمات المحفوظة هي حجزت من قبل مصممي اللغة وتستخدم من الغاية التي حجزت من أجلها.

بعض منها:

Auto, break, case, char, const ,continue, default , do, doubl, else, enum,
long, extern, float, for, goto , if , int , register , return , short , signed , sizeof,
static , struct , switch , typedef , union , while , volatile , unsigned , voil

asm	catch	class	delete	friend	inline
new	operator	private	public	template	this
protected	throw	try	virtual		



Type	size (byte)	min (value)	max(value)
int	4	-2147483648	2147483647
short int	2	-32768	32767
long int	4	-2147483648	2147483647
unsigned int	4	0	4294967295
unsigned short int	2	0	65535
unsigned long int	4	0	4294967295
char	1	-128	127
signed char	1	-128	127
un signed char	1	0	255
float	4	1.175494351e-38F	3.402823466e+38F
double	8	2.2250738585072014e-308	1.7976931348623158e+308

تعرف بالمعطيات الأولية:
تختلف حجوم الأنواع وفق البيئة المستخدمة والمذكورة جانبًا
بيئة Visual Studio 10

المعطيات الأولية (float, double, (char, bool

. string . توجد أربعة أنواع من الثوابت: صحيحة، حقيقة، محرفيه، "متعددة المحارف".

الثوابت الصحيحة والحقيقة تتميز - بعدم وجود فراغات ضمنها. - يمكن أن تُسبق بإشارة. - لا يمكن أن تتعدي قيمة ثابتة أو أن تقل عن حد معين، أي لها حد علوي وسفلي(موضح بالجدول السابق).

أ-الثوابت الصحيحة: أعداد صحيحة يمكن أن تمثل بالنظام العشري والتثمني والساداسي عشر.
ب-الثوابت الحقيقة: يحوي العدد أس عشري أو نقطة عشرية أو كليهما ونذكر بعض الأمثلة.

0. 1. 0.7 320.1 1.66E+8 166E-2 3E-8

يجب أن يكون الأسس عدد صحيح لا يملك فراغات ولا فاصلة عشرية أو عادية...

ت-الثوابت المحرفيه : هي رمز واحد بين فاصلتين علويتين ''\$', '\$' , 'A' , 'a' ويمكن لكل حرف أن يأخذ قيمته من جدول.

American Standard code for Information Interchange (ASCII).

ج- ثوابت سلسلة: توضع الرموز المتماثلة بين إشارتي تصيص مزدوجة معبرة عن السلسل.

سلسل الهرب



ASCII	سلسل الهرب	ال فعل
007	\a	لسماع الجرس (beep)
008	\b	لانانتقال إلى الخلف
009	\t	لانانتقال جدول أفقي
011	\v	لانانتقال جدول رأسي
010	\n	لانانتقال إلى سطر جديد
012	\f	تغذية صيغة الورقة
013	\r	عودة العربية إلى بداية السطر الحالي carriage return
034	\"	لطباعة علامة تصيص مزدوجة
039	'	لطباعة علامة تصيص مفردة
063	\?	لطباعة علامة استفهام
092	\/\	لطباعة شرطة مائلة للخلف
000	\0	لطباعة صفر

سلسل الهرب لبعض الرموز التحكمية والتي تبدأ بالشحطة المائلة \ التي تدعى بحرف escape character والتي تدل على وجود عمل خاص يجب فعلة ثم يليها حرف أو رمز.

[2]



المتغير: يمثل نوعاً واحداً من المعطيات ضمن البرنامج (معطيات أوليه)، عددي أو حرفي أو منطقي، وتملك هذه المتغيرات قيمة واحدة بسيطة من نفس نوعها في اللحظة الواحدة.

6-2-2- المصفوفات arrays

المصفوفة: عبارة عن تجمع عدد من القيم المتساوية النوع ضمن نفس الاسم، والعنصر الواحد هو عنصر مصفوفة وتميز العناصر عن بعضها بعضاً بتمايز دليل المصفوفة الدال عليها.
معظم لغات البرمجة يبدأ الدليل من 0 وينتهي مع $n-1$ إذا احتوت المصفوفة n عنصر ويمكن أن تكون من أي نوع معطيات وأن تكون بعد واحد أو أكثر.



7-2-2 التعابير expressions

تملك قيمة فردية عددية، محرفية أو عنصر مصفوفة أو إشارة تابع ويمكن أن تكون خليط من ذلك تتصل مع بعضها بمعاملات ويمكن أن تكون نتيجة التعبير قيمة واحدة.

كمثال النسب $x=y$ ، عملية حسابية $e=a+b$ ، عملية منطقية $x==y$ نتيجة لها قيمة منطقية حيث $i=1$ وفق العباره $i+1$ يدعى بالمؤثر الأحادي.

8-2-2 العبارات statements

أ-عبارة بسيطة **simple statement**: عبارة عن نتيجة التعبير والتي يليها الفارزة مع النقطة ;

ب-عبارة مركبة **complex statement**: عند الحاجة لتجمیع عدة عبارات بسيطة مع بعضها بعضًا لتعامل كعبارة بسيطة وتوضع هذه العبارات بين {} تصبح عبارة مركبة.



9-2-2. الثوابت الرمزية symbolic constants

الثابت الرمزي: هو اسم يعوض به عن تسلسل من الرموز التي يمكن أن تمثل ثابتاً عددياً أو ثابتاً حرفيأً أو ثابت سلسلة، وعليه يسمح الثابت الرمزي بظهور اسم بدل عنه ويستبدل كل وجود للثابت ضمن البرنامج بتسلسل الرموز الخاص به.

تعرف الثوابت الرمزية في بداية البرنامج وبعدها تظهر في أي مكان نحتاج إليها ويعرف شكلها العام:

#define identifiername value

حيث **define** كلمة مفتاحية للتعريف، **identifiername** الاسم الرمزي، **value** قيمته.
وتكون أحد أنواع المعطيات مثلاً **text** يمثل تسلسل الحروف المصاحبة للاسم الرمزي ولا يملك ; لأن ذلك لا يمثل عبارة C++ صحيحة وإذا وجدت ستكون جزء من الثابت المستخدم كمثال تعريف **pi**.

define pi 3.141593



C++ - 3-2 معمالت لغة Operators of C++

تستخدم المعمالت، الحسابية، والعلائقية، والمنطقية، وعمولات التخصيص أو النسب، والشرطية في تكوين التعبير أو العبارات البرمجية وتأثر المعمالت على العوامل أو العناصر وقد تتطلب عامل أو أكثر. وقد تكون أحادية، ثنائية، ثلاثية.

1-3-2 العمليات الحسابية

العمليات الحسابية المعروفة كالجمع Addition (+) والطرح Subtraction (-) والضرب Multiplication (*) والقسمة Division (/) وبباقي القسمة Module (%).

تستخدم المعملات الحسابية عاملين وفي حالتي القسمة وبباقي القسمة يجب أن يكون العامل الثاني مخالفًا للصفر، كما أن عامل بباقي القسمة يجب أن يكونا صحيحين.



نذكر عدد من الملاحظات

- ✓ يمكن أن نقوم بضرب حرفين ببعضهما البعض أو إجراء أية عملية حسابية عليهما، حيث س يتم التعامل معهما وفق قيم ترميز ASCII لهما.
بفرض أن $e1, e2$ متغيران يمثلان T, P على التوالي وترميز الأسكنى للمحرف P هو 80 في حين الترميز للمحرف T مساوي لـ 84 وبالتالي فإن: ; $e1+e2 \rightarrow 164$ و $e1+e2+5 \rightarrow 169$ و $e1+e2+'5 \rightarrow 217$; حيث أن ترميز ASCII لـ ' 5 ' هو 53.
- ✓ إجراء عملية على عددين صحيحين نحصل في الناتج على عدد صحيح ليكن $b = 3, a = 10$ فإن $a + b \rightarrow 13$; $a - b \rightarrow 7$; $a / b \rightarrow 3$ $a \% b \rightarrow 1$;
- ✓ في إجراء باقي القسمة تحدد إشارة الناتج من إشارة أول عامل، وبفرض:
 $a + b \rightarrow 8$, $a - b \rightarrow 14$, $a * b \rightarrow -33$, $a / b \rightarrow -3$, $a \% b \rightarrow 2$
وعندما تصبح $-11 = a / b$ فإن $a = 3, b = -11$. $a \% b = -2$.
- ✓ عند إجراء عملية على عاملين محددي الإشارة و بدقة مختلفة يتم العمل وفق الدقة الأفضل (الأكبر).
- ✓ يمكن أن يذكر أحد أنواع المعطيات بين قوسين قبل العبارة ليتم تحويلها بشكل ضمني (implicit casting) ويدعى بالتحويل القسري (casting) إلى هذا النوع .

أولويات التنفيذ

Parentheses

1- الأقواس

Multiplication, Division, Modulus

(/ , * , %) -2

Addition, Subtraction

(+ , -) -3

وإذا تساوت الأولويات يتم التنفيذ من اليسار باتجاه اليمين.

مثال:

$2*((i\%5)*(4+(j-3)/(k+2)))$

7 3 6 5 1 4 2



1. تتعامل هذه المعاملات مع عامل واحد وقد تسبق العامل عادةً أو ربما تكون خلفه في بعض الحالات، أكثر هذه المعاملات شيوعاً هو المؤثر السالب، ويسبق الأعداد والثوابت العددية والعبارات الحسابية.
2. مؤثر الزيادة ++ ومؤثر النقصان -- وهما يؤثران بالزيادة أو النقصان بالقيمة 1 وقد تسبق العامل وتكون لها الأولية وقد تليه ويكون لبعض العمليات أولوية عليها.

$m \rightarrow 5 ; m++ \rightarrow 5; m \rightarrow 6$

$n \rightarrow 3 ; ++n \rightarrow 4 ;$

$q \rightarrow 4 ; --q \rightarrow 3 ;$

$p \rightarrow 5 ; p-- \rightarrow 4 ;$

والمعاملات الأحادية تملك أولوية على المؤثرات الحسابية.



3-3-2- المعاملات العلائقية

Relational operators

معاملات المقارنات

التسمية		C++	الرمز الرياضي
Greater than	أكبر	>	>
Less than	أصغر	<	<
Greater or equal than	أكبر يساوي	>=	≥
Less or equal than	أصغر يساوي	<=	≤
Equal	يساوي	==	=
Different ,not equal	لا يساوي	!=	≠

قد تكون العوامل جمل أو تعبير تعطي في النهاية إلى قيم منطقية مثلاً:

$(i+f) <= 10 ; \quad c >= 10 * (i+f) ;$



4-3-2-المعاملات المنطقية

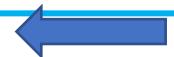
Logical operators

a	b	<code>a&&b</code>
0	0	0
0	1	0
1	0	0
1	1	1

a	b	<code>a b</code>
0	0	0
0	1	1
1	0	1
1	1	1

التابعين المنطقيين `or`, `and` و `||` و `&&`

Op	asm	Description
<code>&</code>	AND	Logical AND
<code> </code>	OR	Logical OR
<code>^</code>	XOR	Logical exclusive OR
<code>~</code>	NOT	Complement to one (bit inversion)
<code><<</code>	SHL	Shift Left
<code>>></code>	SHR	Shift Right



أولوية بعض معاملات لغة C++

Priority	Operator	Description	Associativity
4	* / %	arithmetical operations	Left
5	+ -	arithmetical operations	Left
6	<< >>	bit shifting (bitwise)	Left
7	< <= > >=	Relational operators	Left
8	== !=	Relational operators	Left
9	& ^	Bitwise operators	Left
10	&&	Logic operators	Left
11	?:	Conditional	Right
12	= += -= *= /= %= >>= <<= &= ^= =	Assignation	Right
13	,	Comma, Separator	Left

Priority	Operator	Description	Associativity
1	::	scope	Left
2	() [] -> . sizeof		Left
3	++ --	increment/decrement	Right
	~	Complement to one (bitwise)	
	!	unary NOT	
	& *	Reference and Dereference (pointers)	
	(type)	Type casting	
	+ -	Unary less sign	



5-3-2 عوامل التخصيص أو النسب

Assignment operators

أكثر عوامل التخصيص شيوعاً هو = وله الشكل: identifier = expression; حيث أن identifier هو متغير بصورة عامة أما expression فهو متغير أو تعبير أو خليط منهما.
مثلاً:

```
a=3;      delta =0.001 ;      area =length *width;  
i = j = 5 ;      i = j/2 →2,    i=2* j/2 →5     i = 2* ( j/2 ) →4
```

تملك لغة C++ خمس مؤثرات للتخصيص = ، / = ، * = ، - = ، + =
expression1 + = expression2; → **expression1 = expression1 + expression2 ;**
a *=5 → **a=a*5**



تكتب بصيغتين: الأولى

```
if(expression1 ) Stetment1;  
else Stetment2 ;
```

الثانية: **expression1 ? Stetment1 : Stetment2;** ويرمز له **:** **?**
هنا يتم التساؤل هل قيمة **expression1** هي **true** أم **false** فإن كانت **true** سيتم أخذ التعبير **Stetment1** وتتجاهل **Stetment2** وإذا كانت تساوي **false** سيتم أخذ التعبير الأخيرة **Stetment2** وتتجاهل **Stetment1** ومثال ذلك.

i=(i < 0) ? 0 : 100

إذا كانت **i < 0** أي التعبير صح سوف يتم إسناد القيمة **0** للمتغير **i** وغير ذلك سيتم إسناد القيمة **100** له.

min = (f < g) ? f : g

إذا كانت **f** أصغر من **g** سيكون **min = f** وإلا **.min = g**



بعض الأسئلة المطلوب الإجابة عن التالي:

- 1 - ليكن المطلوب تعريف لغة الآلة، لغة التجميع، لغات عالية المستوى، المفسر والمطابق.
- 2 - ذكر قواعد تسمية المعرفات.
- 3- ذكر أنواع بالمعطيات الأولية.
- 4- تعداد العمليات الحسابية والمنطقية المعروفة في C++.
- 5-ذكر خمس من أولويات التنفيذ للمعاملات.
- 6- ذكر مؤثرات التخصيص مع مثال عن كل منها.

