Ministry of Higher Education and Scientific Research

Al-Mustaqbal University College

Computer Engineering Techniques Department

C/C++ programming language

First year

Lecture (1)

Prepared By

MSc. Zaid Ibrahim Rasool

2020-2021

الفصل الاول

اساسيات لغة C++ (fundamentals of C++)

1.1 الرموز (character)

تتالف الرموز من

1- الارقام (digits) (0,1,2,3,4,5,6)
2- الحروف الانكليزية الكبيرة A,B,C,D,E
3- الحروف الانلكيزية الصغيرة a, b, c, d, e
4- الرموز الخاصة (special characters) وهي

+	/	(	#	=	%	!
_	.	)	;	>	}	"
*	:	$	,	<	{	'

عمليات الاحرف تمثل داخل الحاسوب بواسطة ارقام صحيحة وفقا لنظام(American (ASCII Standard Code for Information International تسمى الاعداد الترتيبية (ordinal numbers) لذا فان الممبرمج بامكانه ان يمزج بين الرموز والاعداد الصحيحة بتعابير رياضية لتؤدي غاية معينة, مثال على ذلك :

اذا فرضنا ان المتغير الرمزي (ch) هو متغير من نوع حروف وتم اسناد قيمة له كما ياتي

(ch='y')

فان التعبير التالي ; ch = ch + 1

ستكون قيمة المتغير الرمزي (ch) تساوي الرمز ('z')

وكذلك فان التعبير التالي ch = ch – 3

ستكون قيمة المتغير الرمزي (ch) تساوي الرمز ('w') وهذا يعتمد على القيمة الرقمية التي تمثل الاحرف بنظام ال (ASCII) ويصرح عن الرموز بلغة البرمجة C++ في اي مكان داخل جسم البرنامج بالمعرف (char) التي تسبق المتغيرات.

1.2 المعرفات (identifiers)

وهي التي تسمح بتعريف الحاسوب بما نريد, المعرف هو ذلك الاسم الذي يخزن به قيمة الثابت او المتغير او الدالة ,وتطلق الاسماء التعريفية على:

A- المتغيرات.

B- الدوال.

C- المؤشرات.

للمعرفات في لغة C++ شروط او قواعد يجب الاتزام بها وهي

a- ان يكون الاسم من حرف او سلسلة من الحروف او الارقام او ان يبدء ب (_) .

b- ان لايحتوي على اي رمز خاص ما عدى الرمز (_) under score .

c- ان لايكون الاسم احدى الكلمات المحجوزة .

d- يسمح له باستخدام الحروف الصغيرة والكبيرة .

ومن الجدير بالذكر فان لغة C++ تميز بين الحروف الكبيرة والحروف الصغيرة فمثلا الاسماء : SYSTEM,System ,system تعامل كأسماء مختلفة عن بعضها البعض بسبب اختلاف معاملة المترجم للحروف الصغيرة والكبيرة .

مثال: لبعض المعرفات الصحيحة

,sum, Door12 , Soft_fine ,A , Ok_, Matrix, X_ray , degree, B6 _new

مثال: لبعض المعرفات الغير صحيحة

7-up ..... لانه بدء برقم وليس بحرف

$cost ..... يبدا بالرمز الخاص($)

Z6.1 .....لاستعماله الرمز الخاص (.)

ZAID IBRAHIM ..... مستخدم فراغآ

Do ..... يمثل كلمة محجوزة

ان اختيار المتغير من قبل الامبرمج تعد مسئلة مهمة ويفضل ان يعكس المتغير المعنى الذي يستخدم لاجله المتغير فمثلا يفضل استخدام المتغير (sum) مع الجمع او (avr) مع المعدل واذا ما استخدم متغير اخر فان ذلك لايؤدي الى خطأ, وكذلك يفضل ان لايكون المتغير طويل مثلا يفضل استخدام متغير من حرف واحد عندما نستخدمه في برنامج قصير ولايتكرر كثيرا.

1.3 الكلمات المحجوزة (Reserved words)

وهي عبارة عن كلمات قياسية موجودة في مكتبة لغة C++ وتكتب عادة بحروف صغيرة ولها وظيفة معينة ولايمكن استعمالها بغير ماوضفت به لان ذلك سوف يحدث ارباك للمترجم (**Compiler**) ولها معاني خاصة تؤديها في برنامج C++ وهذه الكلمات هي:

asm	auto	bool	break	case
continue	default	delete	do	double
false	float	for	friend	goto
operator	private	protected	public	register
struct	switch	template	this	throw
unsigned	using	virtual	void	volatile
catch	char	class	const	cast
else	enum	export	explicit	extern
if	inline	int	long	new
return	short	signed	sizeof	static
true	try	typedef	typename	union
Wchar_t	while	and	or	xor

1.4 الاعداد (Numbers)

الاعداد في لغة C++ هي:

A. الاعداد الصحيحة (Integer) :هي كل الاعداد الموجبة والسالبة والخالية من الفاصلة العشرية, فالصفر هو عدد صحيح و 134 هو عدد صحيح و 66- هو عدد صحيح اما 122.33 و -11.33 فهي ليست اعداد صحيحة.

العلاقات الرياضية التي تستخدم مع الاعداد الصحيحة هي (% , / , * , - , +) وهي على التوالي (الجمع , الطرح , الضرب , القسمة , وباقي القسمة).

امثلة عن الاعداد الصحيحة

-33 , 10000 , 14 , 0

والاعداد التالية تعتبر غير صحيحة لكل من الاسباب التالية

3.33 ..... لانه يحتوي على الفاصلة العشرية .

2,0000 ..... لانه يحتوي على الفارزة .

Z12 ..... لانه يحتوي على حرف ابجدي.

2 4 ..... لوجود فراغ بين العددين .

و يصرح عن الاعداد الصحيحة بلغة C++ في اي مكان داخل جسم البرنامج بالمعرف (int) والتي تعني (integer) وهي تكتب قبل المتغيرات , مثال

int z ;

B. الاعداد الحقيقية (**Real Numbers**) : وهي الاعداد التي تحتوي على الكسور او هي الاعداد التي تحتوي على الفاصلة العشرية مثل

(0.03, 12.5, -356.67890, 10.0).

العمليات الرياضية التي يمكن اجرائها على الاعداد الحقيقية فهي

(/ , * , - , +) وهي على التوالي (الجمع , الطرح , الضرب , القسمة) . يصرح عن الاعداد الحقيقية في اي مكان داخل البرنامج بالمعرف (float) التي تسبق المتغيرات مثل

float z ;

C. الاعداد ذات الدقة المضاعفة : وهي الاعداد الحقيقة الممثلة بشكل قوة اسية باستخدام الرمز (e) والذي يمثل عشرة مرفوع الى اس معين (الاس هو الرقم الذي يلي الحرف (e)) الرقم الذي يلي (e) يجب ان يكون عددا صحيحا), مثل

3.14159 // = 3 .14159
6.02e23 // = 6.02 x 10^23
1.6e-19 // = 1.6 x 10^ -19
3.0 // = 3.0

0.22e9 // = 0.22 x 10^9

1.5الحروف والسلاسل (Characters):

الحرف :

كما ذكرنا سابقا بأن الاحرف تمثل داخل الحاسوب بواسطة ارقام صحيحة وفقا لنظام (ASCII) تسمى الاعداد الترتيبية, حيث يوضع بين علامتي التنصيص المفردة (' ') مثال على ذلك '+' , 'Z' , 'z' , '*' , '7'

السلسلة :

وهي مجموعة من الحروف او الارقام او الرموز الخاصة بشرط ان تكون موضوعة بين علامتي التنصيص المزدوجة (" ").

1.6 المتغيرات (Variables)

وهي اسماء رمزية (عناوين) يخصص لها مواقع في ذاكرة الحاسوب والتي تتغير قيمتها من نوع الى اخر.

وبما ان انواع المعلومات المراد تخزينها تكون عادة مختلفة مثل القيم الصحيحة او الرمزية او الحقيقية لذلك يجب ان يعلن عن المتغيرات مسبقا (اي قبل تخصيص قيمة لها), لغرض التعرف عليها من قبل المترجم (Compiler) ولغرض الاعلان عن قيمة المتغير نستخدم الصيغة الاتية :

Type Variable Name;

حيث يمثل **Variable Name** اسم المتغير

Type يمثل نوع المتغير بعض انواع ال Type الشائعة في الاستخدام داخل البرامج (int, float, char).

فمثلا في المثال التالي :

```
float x,y;
integer k;
char a,b,c;
x=3.0;
y=10.5*x;
k=100;
a=t;
b=f;
c=z;
```

لاحظ السطرين الاول والثالث يوجد اكثر من متغير حيث يفصل بين كل متغير ب (,) فاصلة ,
ونلاحظ ايضا هنالك تحديد بانواع المتغيرات من خلال ذكر التفصيل لكل نوع من حيث طول السعة التخزينية.

اما لغرض اعطاء قيمة للمتغير, فيتم من خلال ما ياتي :

1- جملة التخصيص (=).

الصيغة العامة هي Variable Name=Variable Value;

حيث Variable Value تمثل قيمة المتغير.

2- جملة الادخال (cin>>) .

الصيغة العامة هي cin>> Variable Name;

عندما يصل البرنامج الى جملة cin>> يتوقف منتظرا ادخال قيمة للمتغير عن طريق لوحة المفاتيح, ثم تخزن تلك القيمة في عنوان المتغير المخصص
له في الذاكرة لتصبح Variable Name=Variable Value;

ملاحضات مهمة (important not's)

1- من الممكن الاعلان عن اكثر من متغير لنفس النوع بسطر واحد, و كما يلي :

Example

```
int a,b;
float x,y,z;
a=22;
b=30;
x=0.0;
y=7.5;
z=100.6;
```

2- يمكن اعطاء قيمة لمتغير اثناء الاعلان عن ذلك المتغير وكما يلي:

Example

```
int x= 100 ;
Float a,b= 55.8 ;
Char z=' T ';
```

3- من الممكن استخدام عبارة ادخال مرة واحدة فقط لادخال اكثر من متغير, وكما يلي :

Example

```
int x;
float number;
char z;
cin>> x >> number >> z;
```

1.7 التعابير المنطقية (The Boolean Expressions)

وهي التعابير التي تتمثل نتيجتها ب (صح او خطأ) (true or false) التعبير المنطقي يعيد قيمة (1) عندما يكون التعبير (true) ويعيد قيمة (0) عندما يكون التعبير (false).

Example

```
bool c= (a==b);
```

نلاحظ اننا استخدمنا علامة المساواة على ان النتيجة في الطرف اليمن ستكون هي قيمة الطرف الايسر وقد استخدمنا علامة (==) والتي تستخدم لفحص عملية المساواة, فاذا كانت قيمة (a,b) متساويتان فان قيمة (c) ستكون قيمتها (true) و بخلاف ذلك تكون قيمتها (false).

1.8 المؤثرات (Operators)

1.8.1 العمليات الحسابية (Arithmetic Operators)

يمكن تمثيل العمليات الحسابية بالجدول التالي

المؤثر	معناه	مثال
+	addition	a+b
-	subtraction	a-b
*	multiplication	a*b
/	division	a/b
%	modulo	a%b

يجب الاخذ بنظر الاعتبار بالنسبة للمؤثر باقي القسمة (%) يجب ان تكون القيم المستعملة هي قيم صحيحة والا النتيجة تكون خاطئة.

مثال على ذلك اذا كان z=11 و i=3 فان النتيجة a%b=2,

اما اذا كان كل من a or b او كلاهما عدد حقيقيا, فان عملية باقي القسمة لايمكن تطبيقها, لان النتيجة تكون خاطئة.