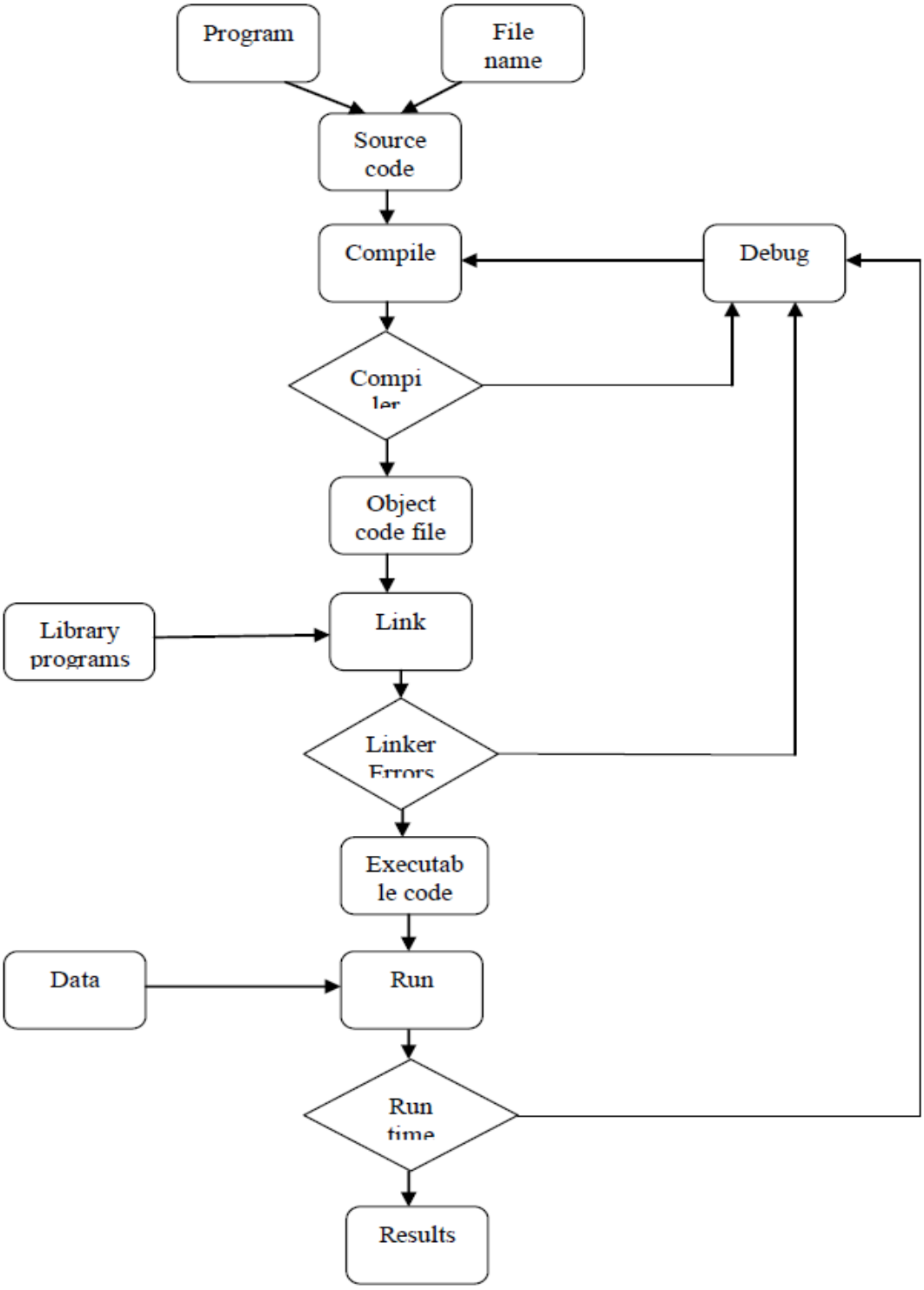
الفصل الاول
أساسيات البرمجة بلغة C++

أمكانية تنظيم ومعالجة البيانات هو مفتاح النجاح في الحياة الحديثة .صمم الحاسوب لحمل ومعالجة كميات كبيرة من المعلومات بسرعة وكفاءة .بشكل عام فان الحاسوب لا يمكنه عمل أي شيء مالم يتم أخباره ما يجب أن يقوم به .لذلك وجدت C++ . C++ هي لغة برمجة عليا(أي قريبة من لغة الإنسان وفهمه)والتي تسمح لمهندس البرامجيات بالتواصل بكفاءة مع الحاسوب .وتعد لغة C++ من اللغات ذات المرونة العالية والقابلة للتكيف .ومنذ اختراعها في عام 1981 فقد تم استخدامها لبرامج واسعة ومختلفة تضمنت تعليمات مخزنة على الحاسوب للمسيطرات الدقيقة (micro controller) ، أنظمة التشغيل (operating systems) ، التطبيقات (applications) ، برامج الرسوم (graphics programs) ، وأصبحت C++ بسرعة لغة البرمجة التي يتم اختيارها.

صممت C++ كجسر بين المبرمج والحاسوب .الفكرة بجعل المبرمج ينظم البرنامج بطريقة (هو /هي) يفهمه بسهولة .بعدها يقوم المترجم (هو برنامج واجبه تحويل اللغة العليا إلى اللغة التي يتعامل بها الحاسوب) بنقل اللغة (البرنامج) إلى صيغة تستطيع الماكنة استخدامها (التعامل معها) برنامج الحاسوب يتكون من جزئين :هيكل البيانات والايعازات .يفرض الحاسوب او لا يفرض القليل من التنظيم على هذين الجزئين .بعد هذا كله فان الحواسيب مصممة لان تكون عامة قدر الإمكان .البيانات في الحاسوب تخزن كسلسلة من البايتات و C++ تنظم هذه البايتات ببيانات مفيدة .الإعلان عن البيانات تستخدم من قبل المبرمج لوصف المعلومات التي (هو/هي) يتعامل معها.

برامج C++ تكتب بلغة عليا باستخدام الأحرف، الأرقام، والرموز الأخرى التي نجدها على لوحة المفاتيح . واقعا فان الحواسيب تنفذ البرامج المكتوبة بلغة دنيا تدعى لغة الماكنة (machine language) (والتي هي سلسلة من الأرقام ممثلة بطريقة الصفر، واحد). لذلك، وقبل ان يتم استخدام البرنامج يجب أن يكون هنا عدد من التحويلات .البرامج تبدأ كفكرة في رأس المبرمج .يقوم المبرمج بكتابة أفكاره في ملف، يدعى ملف المصدر (source file or source code) مستخدما محرر اللغة. هذا الملف يحول بواسطة المترجم الى (الملف الهدف) (object file) بعدها يستدعي البرنامج الرابط (linker) حيث يأخذ الملف الهدف ليربطه أو يشركه مع روتينات معرفة مسبقا من المكتبة القياسية (standard library) لينتج برنامج قابل للتنفيذ (والذي هو عبارة عن مجموعة من أيعازات لغة الماكنة).)الشكل (1) يبين خطوات تحويل البرنامج المكتوب بلغة عليا إلى برنامج قابل للتنفيذ.



الشكل (1) خطوات تنفيذ البرنامج بلغة C

2-4 خصائص لغة C++

C++ هي اللغة الأكثر استخداما في العالم .هذه اللغة لها صفات وخصائص تميزها عن لغات البرمجة الأخرى، وأكثر هذه الصفات هي:

أ- البرمجة الكيانية : إمكانية تصميم البرنامج على شكل كيانات تسمح للمبرمج تصميم تطبيقاته، لتكون أكثر اتصال بين الكيانات بدلا من هيكل الشفرة المتتالية .فضلا عن أنها تسمح بامكانيه كبيرة إلى أعادة استخدام الشفرة بطرق أكثر منطقية وإنتاجيه.

ب- النقل : بامكانك عمليا ان تترجم نفس شفرة C++ على الأغلب في أي نوع من الحواسيب وأنظمة التشغيل دون أجراء تغييرات صعبة.

ت- الإيجاز : الشفرة التي تكتب بلغة C++ هي قصيرة جدا بالمقارنة مع اللغات الاخرى، حيث يفضل استخدام الرموز الخاصة للكلمات المفتاحية، وهذه تختزل بعض الجهد المبذول من المبرمج.

ث- برمجة الأجزاء: من الممكن ان تكون تطبيقات C++ من عدد من الملفات لشفرة المصدر والتي تترجم بشكل منفصل، ثم يتم ربطها مع بعض، هذا يساعد على تقليل الوقت وليس من الضروري اعادة ترجمة كامل التطبيق عندما يتم عمل تغيير مفرد ولكن فقط الملف الذي يحتويه .بالإضافة لذلك، فان هذه الخاصية تسمح لربط شفرة C++ مع الشفرة الناتجة بلغات أخرى مثل المجمع (assembler) أو C.

ج- التوافق مع لغة C: C++ هي البوابة الخلفية للتوافق مع لغة C ، أي شفرة تكتب بلغة C سيكون من السهولة تضمينها في برنامج C++ دون الحاجة لأي تغييرات صعبة.

ح- السرعة : الشفرة الناتجة من تجميع C++ هي كفوءة جدا، وذلك بسبب كونها لغة ثنائية،فهي تعد من اللغات ذات المستوى العالي ومن اللغات ذات المستوى الواطئ فضلا عن صغر حجم اللغة نفسها.

3-4 أجزاء برنامج لغة C++

يكون تسلسل كتابة أجزاء البرنامج بلغة C++ حسب ما يلي:

1- عبارة التضـمين (#include) : يتم في هذا الجزء اسـتدعاء الدوال المطلوب التعامل معها في البرنامج . يتوفر بلغة C++ مجموعة من الدوال المكتبية مقسـمة كل واحدة منها لعمل وظيفة معينه وهذه الدوال موجودة ضـمن المكتبة القياسـية للغة C++ (Standard Library) إذ يذكر أسم الملف (الدالة المراد استدعائها) بعد عبارة (#include) مباشـرة ويكون محدد بين العلامتين< > ومن الأمثلة على هذه الدوال مثل مكتبة دوال القراءة والطباعة ومكتبة الدوال الرياضية ومكتبة الدوال الخاصة بالسلاسل الرمزية .....الخ.

2-الدالة الرئيسية (main) : في لغة البرمجة C++ فأن البرنامج هو تجميع للدوال والبرامج البسيطة تحتوي على دالة واحدة فقط يعبر عنها بكلمة (main) وعادة فأن التنفيذ يبدأ عند (main) وعادة الدالة الرئيسة تعيد قيمة من نوع الإعداد الصحيحة إلى نظام التشغيل، هذه القيمة تكون صفر عند إكمال تنفيذ البرنامج بشكل صحيح وتكون أي قيمة أخرى حسب ما يتم تحديده بعبارة الإعادة التي ستبين لاحقا. يتم كتابة مكونات الدالة الرئيسية بين القوسين المجعدين { } وتشمل هذه المكونات :

أ. الإيعازات (الأوامر أو العبارات (statements)) : وتشمل مجموعة من الإيعازات لأداء وظائف معينه أو دوال قياســـية بلغة C++ مثل (عبارات الإدخال والإخراج- التعابير الرياضـــية والدوال- عبارات القرار-عبارات التكرار وغيرها).

ب. عبارة الأعاده (return) : هذه العبارة هي أخر عبارة تكتب في الدالة الرئيسية وتسمى بعبارة الأعادة أو الرجوع وهي تشير إلى القيمة التي سوف يرجع المترجم لها الدالة عند تنفيذ البرنامج, بحيث هذه القيمة يجب أن تكون من نفس النوع البياني (Data Type) المحدد من قبل (main). قيمة الرجوع من الممكن أن تكون صفر أو أي تعبير أخر تكتب بعد كلمة (return) ، يمكن الاستغناء عن كتابة كلمة (return) في الدوال اللارجوعية أي أن دالتها الرئيسية لا تعود بقيمه يتم ذلك بوضع كلمة (void) قبل (main) والتي تعني لاشيء .

مما ينبغي الإشارة أليه أن كل الأسطر بداخل الدالة الرئيسية يجب أن تنتهي بفارزه منقوطة.

4-4 مكونات لغة C++

4-4-1 رموز لغة C++

*الرموز المستخدمة في لغة C++ :

1 -الحروف الإنجليزية الكبيرة A,B,C

2 -الحروف الإنجليزية الصغيرة a,b,c

3- الأرقام العربية الأصل 1,2,3

4- رموز خاصة مثل:

[]	"	!	<	-	+
*	،	||	>	( )	_
>>	<>	<=	>=	\	/
!=	&	%	$	#	<<

4-4-2 كلمات لغة C++

4-4-2-1 الكلمات المحجوزة

وهي كلمات قياسية معروفة مسبقا لمترجم C++ ويجب أن تكتب بحروف صغيرة ،ولها معان خاصة بها تؤديها في برنامج C++ ولا يجوز استخدامها أو إعادة تعريفها لغير ما خصصت به وعدد هذه الكلمات المحجوزة هو 59 كلمه كما بالجدول أدناه.

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

4-4-2-2 المعرفات (Identifiers)

ويقصد بها الأسماء التعريفية التي تطلق على المتغيرات-الأختزانات (الدوال) – المؤشرات وهناك قواعد لتسمية الأسماء التعريفية في لغة C++:

1) أن يكون الاسم مكتوبا من سلسلة متصلة من الحروف أو الأرقام شريطة أن لا يبدأ برقم.
2) أن لا يحتوي الاسم على رموز خاصة عدا الخط التحتي "_"
3) أن لا يكون أحدى الكلمات المحجوزة.

بعض الأمثلة الصحيحة عن الأسماء التعريفية :

B6	X_Ray	Matrix	Ok_	A
Soft_Fine	Door12	_New	_New	Sum

ومن الجدير بالذكر أن لغة C++ تفرق بين الحروف الأبجدية الصغيرة والكبيرة ، فمثلا الأسماء : SYSTEM , system , System تعامل كأسماء مختلفة عن بعضها البعض بسبب اختلاف معاملة المترجم للحروف الصغيرة والكبيرة.

3-4-4 الاقترانات المكتبية

الجدول أدناه يمثل مجموعة من الاقترانات المكتبية وما يقابلها بلغة C++ والتي يستفاد منها المبرمج بكتابة البرنامج.

الاقتران	الغرض منه
abs(x)	حساب القيمة المطلقة لـ x
sqrt(x)	حساب جذر x
pow(x,y)	حساب x مرفوعة للقوة y
log(x)	حساب اللوغارتم الطبيعي لقيمة (x) أكبر من الصفر
Log10(x)	حساب اللوغارتم العشري ويجب أن تكون (x) أكبر من الصفر
exp(x)	حساب e^x حيث e=2.71828
sin(x)	حساب جيب x
cos(x)	حساب جيب تمام x
tan(x)	حساب ظل x
asin(x)	حساب الجيب المعكوس لـ x
acos(x)	حساب الجيب تمام المعكوس لـ x
atan(x)	حساب الظل المعكوس لـ x
ceil(x)	توجد أكبر عدد صحيح للقيمة x
floor(x)	تعطي القيمة السابقة للعدد x
fmod(x,y)	لإعطاء باقي قسمة x من y
modf(x,&y)	لاستخراج الجزء الكسري من العدد

*** يجب استخدام الدالة (الملف) (Header file) المسمى (math.h) من مكتبة لغة C++ حيث تذكر بعد كلمة (#include) وتحصر أسم هذا الملف بين العلامتين <> وذلك في جميع الاقترانات التي وردت بالجدول أعلاه.**

4-4-4 الأعداد (المتغيرات) في لغة C++

1-4-4-4 الأعداد الصحيحة Integer Numbers

الأعداد الصحيحة هي كل الأعداد الموجبة والسالبة التي لا تحتوي على كسر .فالصفر عدد صحيح و 123 هو عدد صحيح و -45 أيضا عدد صحيح. أن أي محاولة لاستخدام قيم خارج نطاق الحدود العليا والدنيا للأعداد الصحيحة سيؤدي إلى حدوث خطأ. العلاقات الرياضية التي تستخدم مع الأعداد الصحيحة هي (+ , - , * , / ,

(% وهي على التوالي(الجمع، الطرح، الضرب، القسمة، وحساب باقي القسمة) ويكون ناتج هذه العمليات الحسابية صحيح أيضا.

ويصرح عن الأعداد الصحيحة بلغة C++ في أي مكان داخل البرنامج بالمعرف (int) والتي تعني (integer) وهي تكتب قبل المتغيرات. وتصنف الأعداد الصحيحة حسب طولها والسعة التخزينية لها في الذاكرة وكما يلي :

i. أعداد صحيحة قصيرة short int.
ii. أعداد صحيحة طويلة long int.
iii. أعداد صحيحة بدون أشارة unsigned int.

الجدول أدناه يوضح أنواع الأعداد الصحيحة ومدى كل منها والسعه التخزينية لكل نوع :

أنواع البيانات	المدى	الحجم بلبتات
short int	-32767 to 32767	16
unsigned short	0 to 65535	16
long int	-2147483647 to 2147483647	32
unsigned long	0 to 4294967295	32

مثال : أذا كان x هو متغير صحيح يمكن تعريفة بالشكل التالي

```
int x;
```

عند وجود أكثر من متغير صحيح مثل (x,y,z) يمكن ان تعرف بالشكل التالي

```
int  x,y,z;
```

مثال:

```
21 / 3 = 7
9 / 2 = 4
5 % 2 = 1
7 % 4 = 3
```

2-4-4-4 الأعداد الحقيقية Real Numbers

وهي الأعداد التي تحتوي على كسور ومن أمثلتها (10.0 , 356.67890- , 12.5 , 0.03) أما العمليات الرياضية التي ممكن أجراها على الأعداد الحقيقية فهي (/ , * , - , +) وهي على التوالي (الجمع، الطرح، الضرب، القسمة) ونتيجة هذه العمليات الحسابية هي أعداد حقيقية أيضا.

ويصرح عن الأعداد الحقيقية في لغة البرمجة C++ في أي مكان داخل البرنامج بالمعرف (float) التي تسبق المتغيرات الحقيقية. وتقسم الأعداد الحقيقية إلى ثلاثة أنواع نسبة إلى طولها وحجمها كما مبين بالجدول التالي:

نوع البيانات	المدى	الحجم بالبتات
float	$3.4x10^{-38}$ to $3.4x10^{+38}$	32
double	$1.7x10^{-308}$ to $1.7x10^{+308}$	64
long double	$3.4x10^{-4932}$ to $1.1x10^{+4932}$	80

ملاحظات :

1. يفضل عند استخدام التعريف (long) وضع حرف (L) بعد القيمة كما في المثال :

```
long SunDistance = 93000000L ;
```

2. ممكن أن تسبق الأرقام أحد العلامتين (- أو +) للدلالة على كون الرقم موجب أو سالب (يعد الرقم موجبا أذا لم تظهر أي من العلامتين على يساره).

3. يمكن تمثيل الأرقام بالطريقة الآسية وذلك باستبدال الرقم (10) بالحرف e فمثلا العدد ($2.6*10^{-6}$) يعبر عنه بالصيغة الآسية بالعدد (2.7e-6).

4. نتيجة قسمة عدد صحيح على عدد صحيح أخر هو عدد صحيح.أما أذا كان أحد العددين هو حقيقي فأن النتيجة ستكون عددا حقيقيا، مثال :

```
2.0 / 3 = 0.66666667
50 / 2.0 = 25.0
```

5-4-4 المتغيرات الرمزية (Characters)

وهي كافة الرموز التي تستخدم في الحاسوب والتي غالبا ما نجدها على لوحة المفاتيح والتي تشمل الحروف الأبجدية سواء كانت حروف كبيرة (A..Z) أو حروف صغيرة (a..z) ، الأرقام (0..9))، الرموز الأخرى التي نراها على لوحة المفاتيح مثل (/,+,!,%,&,$,@,…) وتستخدم بشكل منفرد شريطة أن يحاط الرمز بالحاصرتين (' '). ويصرح عن المتغيرات الرمزية بلغة البرمجة C++ في البرنامج بسبقها بكلمة (char) حسب المثال التالي:

```
char color;
color='r';
```

6-4-4 التعابير المنطقية The Boolean Expressions

وهي التعابير التي تمثل نتيجتها بحالة واحدة من أثنين وهم (صح أو خطأ) (true or false) أنواع المتغيرات التي تستخدم لهذا الغرض يصرح عنها في حقل المتغيرات بالأمر (bool) يكثر أستخدام التعابير المنطقية في

الجمل الشرطية ، في لغة C++ تعامل النتيجة (false) على انها صفر (0) وتاخذ النتيجة نعم (true) أية قيمة غير الصفر والمشهور أنها (1). ويبين الجدول التالي الأدوات والعلاقات المنطقية بلغة C++ :

الأدوات العلاقية	
الأداة	**معناها**
>	أكبر من
<	أصغر من
>=	أكبر من أو يساوي
<=	أصغر من أو يساوي
==	يساوي
!=	لا يساوي
الأدوات المنطقية	
&&	And (حرف العطف واو)
||	Or (حرف العطف أو)
!	Not (للنفي)

7-4-4 الثوابت constants

الثوابت هي المقادير التي لا تتغير قيمها أثناء تنفيذ البرنامج ويمكن تعريفها باستخدام كلمة const كما بالصيغة أدناه:

const TYPE *variable_name* = *value* ;

مثال :

const double pi = 3.1413926535 ;

كذلك يمكن تعريف الثوابت قبل الدالة الرئيسية (main) في بداية البرنامج باستخدام الموجة (#define) وكما بالصيغة التالية:

#define *variable_name value*

مثال :

#define pi 3.141396535

تتيح لغة C++ استخدام الثوابت الرمزية (String) كما بالمثال أدناه:

const string University = "Al _ Basrah University " ;

عند استخدام الثوابت الرمزية يجب مراعاة استخدام (Header File) المسمى (String) وذلك من مكتبة لغة C++ ويتم ذكر أسم هذا الملف في بداية البرنامج مع الموجه (#include) وذلك بحصرة بين العلامتين <>.

ملاحظة : الثوابت الرمزية يجب أن تحاط بين علامتي الاقتباس (" ") وأي تعبير وان لم يكن متكون من أحرف أو رموز محاط بهاتين العلامتين يسمى تعبير رمزي حتى لو احتوى على عمليات رياضية فان الحاسب يعتبره تعبير رمزي لا يتم احتساب القيم فيه ويطبع على الشاشة كما هو.

8-4-4 الأدوات الحسابية Arithmetic Operators

تسمح لغة C++ باستعمال العمليات الحسابية من جمع، طرح،ضرب،قسمة وغيرها من العمليات الأخرى التي تتيحها هذه اللغة والجدول التالي يصف رمز كل أداة ووظيفة كل منها بلغة C++ :

العملية الرياضية	الرمز
Addition الجمع	+
Subtraction الطرح	-
Multiplication الضرب	*
Division القسمة	/
Modulo استخراج باقي القسمة	%
Increment للزيادة	++
Decrement للنقصان	--

من مزايا لغة C++ أنها تستخدم الأداتين الحسابيتين (++) لأغراض الزيادة بمقدار واحد و (--) لأغراض النقصان بمقدار واحد حيث يستخدم كل من هذين العاملين بطريقتين مختلفتين أما أن يسبق المتغير أو يلي المتغير كما بالمثال أدناه:

```
A++;
++A;
```

كل من التعبيرين مكافئين للصيغة :

```
A=A+1;
```

لكن كلا التعبيرين ليسا متشابهين فكل منهما له معنى خاص فعندما يسبق عامل الزيادة المتغير فان المتغير تزداد قيمته بمقدار واحد ثم يستخدم أما أذا جاء عامل الزيادة بعد المتغير فان المتغير يستخدم حسب قيمته الحالية وبعدها يزاد بمقدار واحد أي أن العملية الأولى هي جمع تقديم والثانية جمع تأخير. أما العامل (- -)

فتعمل بالطريقة نفسها التي يستخدم فيها عامل الزيادة أي قبل وبعد المتغير مع الاختلاف أن استخدامها يقلل قيمة المتغير بمقدار واحد.

9-4-4 أولوية العمليات الحسابية والمنطقية

تعد جميع أدوات الجمع والطرح والضرب وباقي القسمة في لغة C++ أدوات ثنائية (binary) أي أنها تأخذ (تتعامل مع) قيمتين وتنتج قيمه واحده ، فمثلا نتيجة (3*2) هي 6 وهناك أداة أحادية (Unary) تتعامل مع قيمه واحده كما في العدد (1993-) فالعملية هنا ليست عملية طرح وتمثل الإشارة السالبة للعدد.

أن أولوية تنفيذ العمليات كما يجريها مترجم C++ بالنسبة للأدوات الحسابية والمنطقية هي على النحو التالي:

1) محتوى الأقواس.
2) تنفيذ الدوال.
3) أدوات الزيادة والنقصان.
4)عملية الضرب والقسمة.
5) عملية الجمع والطرح.
6) العمليات المنطقية.

الجدول أدناه يبين تسلسل أولوية العمليات ورموزها ونوعها:

رمز الأولوية	الأداة	نوعها
1	+،-،++،--،!	العوامل الأحادية Unary Operators
2	*،/،%	العوامل الرياضية الثنائية The Binary Arithmetic Operations
3	+ ، -	العوامل الرياضية الثنائية The Binary Arithmetic Operations
4	>،<،>= ،<=	العوامل المنطقية The Boolean operations
5	!=،==	العوامل المنطقية The Boolean operations
6	&&	العوامل المنطقية The Boolean operations
7	||	العوامل المنطقية The Boolean operations

كما هو معلوم أن عملية الضرب والقسمة متكافئة لكن عند ورودهما معا في تعبير ما تكون الأولوية للعملية الأقرب إلى يسار التعبير.

4-4-10 تعابير أخرى

I. فضاء الأسماء القياسية (Standard Namespace)

لغة C++ تقسم الأسماء إلى فضاءات أسماء، وفضاء الأسماء هو تجمع للأسماء ، مثل الأسماء (Ali، Mohammed، Basra) العبارة التي تحدد فضاء الأسماء بالطريقة الموضحة ادناه تدعى الموجة (using):

```
using namespace std;
```

يكتب السطر أعلاه بعد جمل التضمين التي تحتوي كلمة (#include) أي بالسطر الثاني . أن الموجه الخاص (using) يفيد أن البرنامج يستخدم أو يفرض استخدام فضاء الأسماء القياسية (std) أي أن جميع الأسماء التي تذكر بالبرنامج ستكون معرفة بفضاء الأسماء القياسية كما سيذكر لاحقا عند الدخول بفصول كتابة البرنامج.

II. أداة حجم البيانات (sizeof)

تعد هذه الأداة أداة أحاديه (unary) ، وتستعمل لتعيين حجم المتغيرات (بالبايت) ، وتختلف أحجام البيانات حسب نوعها ، لذا تستخدم هذه الأداة لهذا الغرض وتستخدم كما في المثال أدناه:

```
a=sizeof(char)
```

في المثال أعلاه فان قيمة (a) ستكون (1) وذلك لان النوع (char) هو نوع بطول بايت واحد .القيمة المعطاة بواسطة (sizeof) ثابتة، لذلك دائما تحسب قبل تنفيذ البرنامج.

III. التعليقات comments

تستعمل جمل التعليق في أي مكان بالبرنامج لإبداء ملاحظة ما ولا تؤثر على مسار تنفيذ البرنامج وتستخدم لغة C++ لهذا الغرض خطين مائلين كما بالمثال أدناه:

```
// you in first class
```

أمثلة

مثال1: بين المقبول من غير المقبول من الأسماء التعريفية التالية بلغة C++ مع بيان السبب لغير المقبول:

b6.1,const,catch, _New,Ok_,B6,first name, next.word, 15may,Ten%

الحل :

Ten% : غير مقبول لوجود الرمز %

15may: غير مقبول لأنه بدأ برقم

next.word : غير مقبول لاحتوائه على فاصله

first name: غير مقبول لاحتوائه على فراغ

B6 : مقبول

Ok_ : مقبول

New_: مقبول

Catch : غير مقبول لأنه كلمه محجوزة

Const : غير مقبول لأنه كلمه محجوزة

b6.1 : غير مقبول لاحتوائه على فاصله

مثال 2: بين كيف تمثيل كل من المتغيرات التالية:

المتغير	قيمته
a	15
b	-1000
X	1.0
z	4.4544
Ov	6.02e+023
distance	93000000
color	R

الحل :

```
int  a,b
float  x;
double z;
double long Ov;
long int distance=93000000L;
char color='r';
```

مثال 3 : عين نوع التعبير من حيث حسابي -رمزي-منطقي في كل مما يلي :

1) "5*2+12/8" رمزي
2) sqrt(x+2)/3 حسابي
3) factor<=1.0 منطقي
4) maximum=1000 منطقي

مثال 3: أذا كانت (b=7) و (a=2) أوجد قيمة C في كل مما يلي :

1) c= a*++b
2) c=a*b++
3) c=a*--b
4) c=a*b--

الجواب:

1) c=2*(7+1)=2*8=16
2) c=2*7=14
3)c=2*(7-1)=2*6=12
4)c=2*7=14

مثال 4: أكتب التعابير الجبرية التالية بلغة C++:

1) $\sqrt{x^2 + y^2}$
2) $|a + b|cd$
3) $cos^2(a + b)$
4) $\tan^{-1}(y/x)$
5) e^{a+b}

الحل :

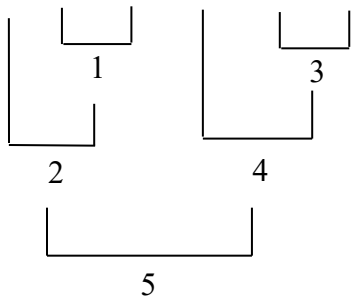
1) sqrt(pow(x,2)+pow(y,2))
2) (abs(a+b))*c*d
3) pow(cos(a+b),2)
4)atan(y/x)
5) exp(a+b)

مثال 5 : بين أولوية العمليات الحسابية في كل التعابير التالية بلغة C++:

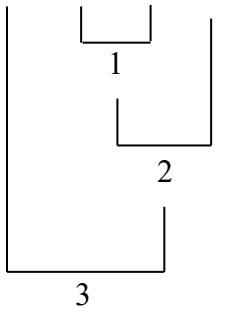
i. x+y/z*a
ii. sin(a+b)-m/sqrt(d)
iii. z= a/b++ - c * d--
iv. a/-b*c

الحل:

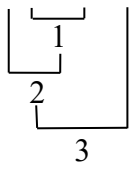
sin(a + b) − m / sqrt(d)



x + y / z * a



a / –b * c



مثال 6: بين السبب في عدم قبول أي من التعابير الحسابية والمنطقية الآتية (أن وجد) لكل مما يلي:

التعبير	سبب عدم القبول
"red"+"blue"	لا يجوز إجراء العمليات الحسابية على التعابير الرمزية
a='x' || b='y'	عدم استخدام أقواس للمعادلتين على جنبي OR
(x/y)>="Ahmed"	لا يجوز مقارنة قيمه حسابيه مع رمزية
(2*a+pow(b,2)	الأقواس غير متقابله
(x+y+z)>=0	مقبول
abs(a-b)+c	مقبول