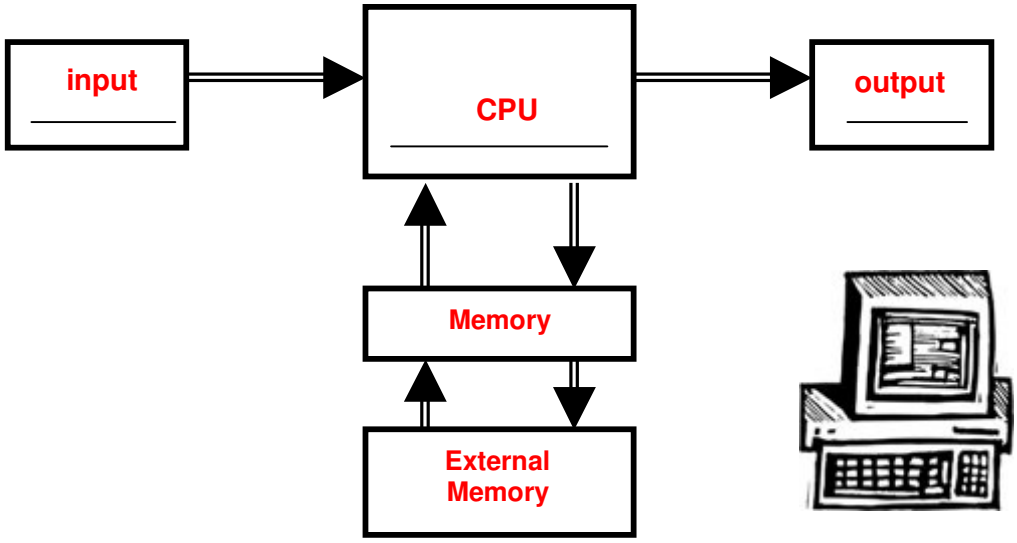
Concepts Review

1. An algorithm is a sequence of steps to solve a problem.

2. A program is the implementation of an algorithm in a particular computer language, like **C** and **C++**.

3. A flowchart is the graphical representation of an **algorithm**.

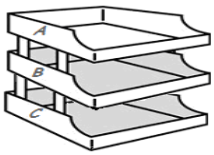
4. The main components of a typical personal **computer system** are:



5. The main elements to make programs are:


Data input

Data output



Variables


Operations


Selection


Loops

6. C/C++ distinguishes several kinds of data:

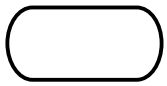
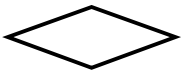
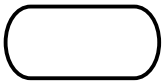
__int_____ is used to handle whole numbers.

__float_____ is used to handle numbers with decimals.

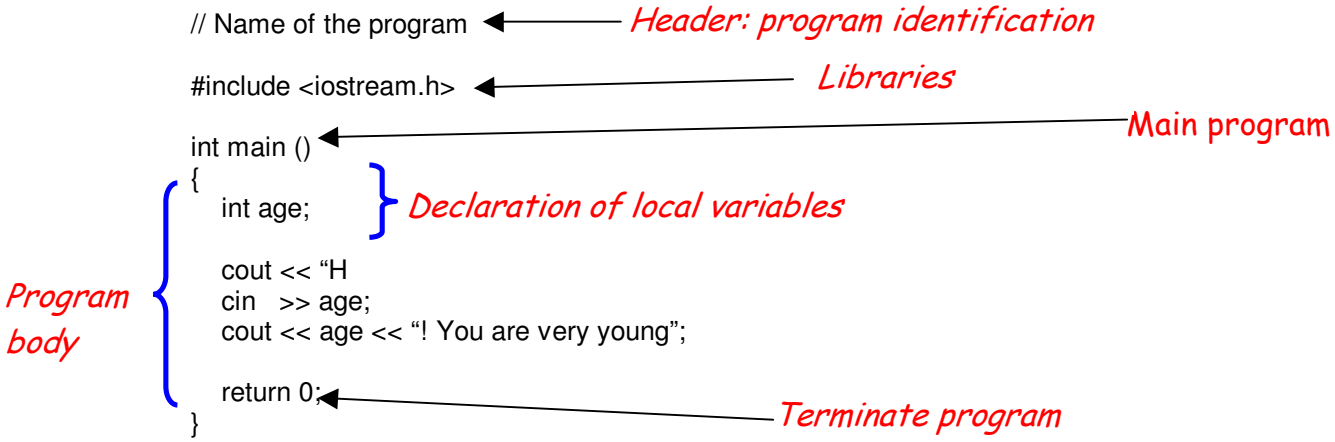
__char_____ is used to handle symbols and letters.

__string____ is used to handle messages.

7. Fundamental notation in C++.

Beginning of instruction block	{		
Input data	cin	>>	
Output data	cout	<<	
Arithmetic operation	y = 2*x - 3;	=	
Conditional statement	if…else	==	
End of instruction block	}		

8. Parts of a C/C++ program.



9. A **question prompt** is the output that precedes any data input.

10. Basic **dialogue** consists of a question prompt and the corresponding data input instruction.

11. Operations in C/C++.

=	+	-	*	/	%	<	>	==	!=	<=	>=	&&	¦¦	!
	+	-	x	÷		<	>	=	≠	≤	≥	and	or	not
Assignment	Arithmetic					Relational						**Logical**		

12. Operator precedence in algebra, C and C++:

( )	* /	+ -
1st	2nd	3rd

13. Let *a*=5, *b*=10, *c*=2, *d*=5 and *e*=1, indicate the value of *y* after **evaluating** the following **expressions**:

a) y = a + b * c / d + e ; // y is _____

__ + __ * __ / __ + __

b) y = a + (b * c) / d + e; // y is _____

__ + (__ * __) / __ + __

c) y = a + b * (c / d) + e; // y is _____

__ + __ * (__ / __) + __

d) y = (a + b) * c / d + e; // y is _____

(__ + __) * __ / __ + __

e) y = a + b * c / (d + e); // y is _____

__ + __ * __ / (__ + __)

f) y = (a + b) * c / (d + e); // y is _____

(__ + __) * __ / (__ + __)

14. A formula can be broken down in several steps to **track** it more easily.

$$X_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

S1= 4*a*c
S2 = b*b
S3 = 2*a
S4 = √ (S2 – S1)
S5 = -b + S4
S6 = -b - S4
X1 = S5 / S3
X2 = S6 / S3

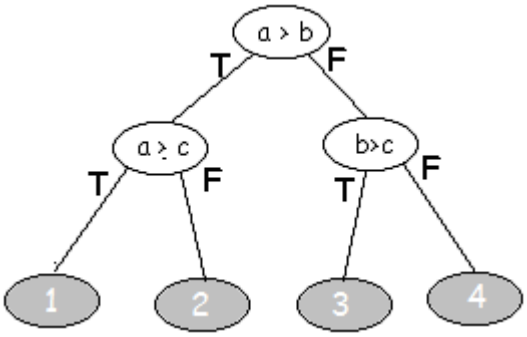
15. A **Boolean expression** compares two values, which results in any of two possible outcomes: true or false.

16. Boolean expressions are used in conditional statements and iterative statements.

17. Given the expression 0<*X*<10, how many values can *X* take?

a) If *X* is an integer: __9_____ b) If *X* is a floating point: _infinite__

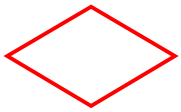
18. A **decision tree** shows all possible outcomes given several Boolean expressions.



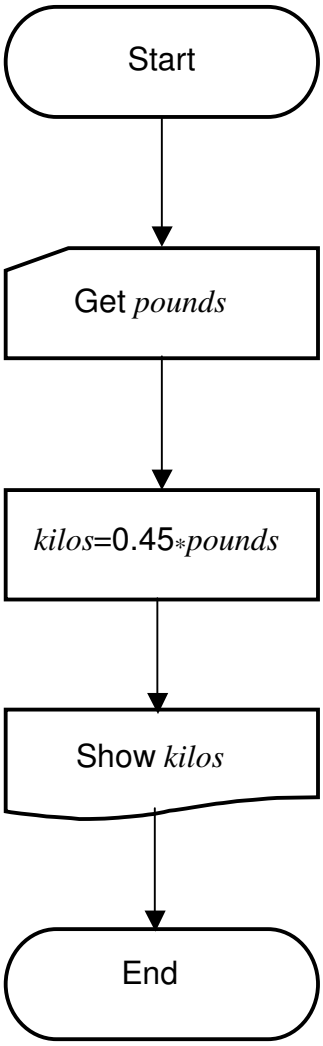
Case 1: a>b, a>=c
Case 2: a>b, c>a
Case 3: a<=b, b> c
Case 4: a<=b, b<=c

19. A flowchart is a step-by-step visual guide indicating how to perform a specific task. The symbols most widely used are:

Start/end task		
Get data from user		

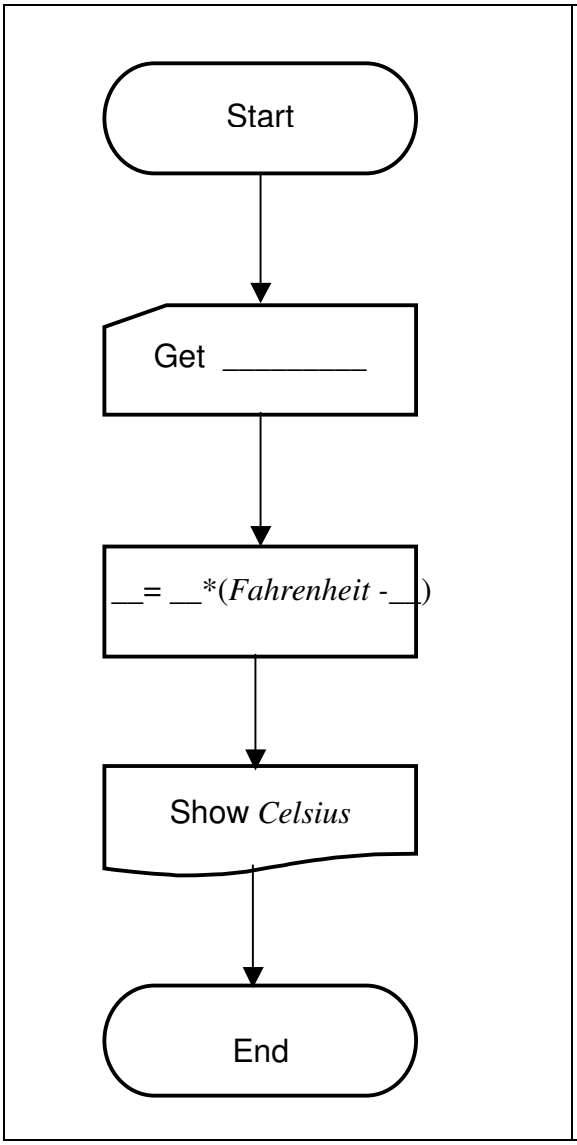
Display results		
Perform operation		
Evaluate condition		

20. An example of a sequential flowchart, and its corresponding C++ program, to transform any amount of pounds into kilograms.




```
#include <iostream.h>

int main () {

float pounds, kilos;


cout << "Enter pounds:";
cin >> pounds;



kilos = 0.45 * pounds;



cout << "Kilograms ";
cout << kilos << endl;



return 0;
}
```

21. An example of a sequential flowchart, and its corresponding C++ program, to transform any amount of Fahrenheit degrees into Celsius.




```
#include <iostream.h>

int main () {

float celsius, fahrenheit;


cout << "Enter degrees:";
cin >> Fahrenheit;




celsius = ____* fahrenheit


cout << "Celsius: ";
cout << fahrenheit<< endl;


return 0;
}
```

22. How many times the message is printed?

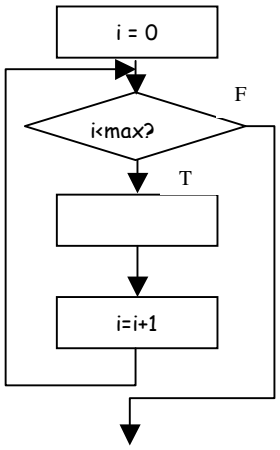
```
for (num=-5;num<=4; num=num+1) {
    cout << "Hello World" << endl;
}
```

a) __10 times_____

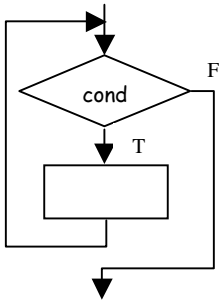
```
while (num < 10) {
    cout << "Hello World" << endl;
}
```

b) _Unsure (either none or infinite)

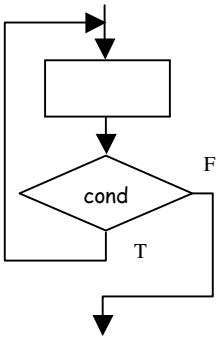
23. An iterative statement repeats a block of instructions several times.



for



while

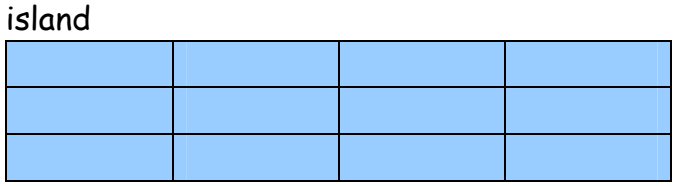


do-while

24. An array is a numbered collection of data of the same type.

25. To declare arrays:

int score[8];



char island[3][4];

row column

26.Given this list:

12	-1	9	5	-4	-6	2	8
0	1	2	3	4	5	6	7

a) The number in the fifth box is __-4_____

b) The tag of the box containing 2 is __6___

27. This piece of code computes the average of an array of 115 elements.

```
avg = 0.0; total = 115;
for (i=0; i < total; i = i + 1) {
    avg = avg + grades[i]; }
avg = avg / total;
```

28. This piece of code displays the contents of a matrix of N X M elements.

```
M=25; N=73;
for (i=0; i < N; i = i +1) {
    for (j=0; j < M; j = j +1) {
        cout << island[i][j]; }
    cout << endl;
    }
```

29. The following program is supposed to **compute** the decimal sum of the values 1/1, ½, 1/3, ¼, 1/5. However, it has several **bugs**:

```
int main () {
     float sum = 0;
     int number = 0;

     while (number < 5) {
         result = 1.0 / number;
         sum = sum + result;
     }
     cout << sum << endl;

     return 0;
}
```

```
int number=1;
float result;
while (number <= 5) {
```

30. After executing this program the screen will show the value: _____.

```
int main () {
     int i,x;

     x =0; i = 0;
     do {
         x = x + i * i;
         i = i + 1;
     } while (i < 10);
     cout << x << endl;

     return 0;
}
```

31. What is the meaning of the term **variable**, within the computing field?

A data storage space (usually an electronic component) associated with an **identifier** (name or symbol) that contains a value, and can be changed throughout the execution of the program.

32. What do you understand for **computing agent**?

    Something (or someone) able to "mechanically" carry out the instructions stated in an algorithm.

33. What is a **conditional statement**?

    It is an instruction that evaluates a **Boolean expression** in order to execute a specific action.

34. What is the truth-value of a conditional statement?

    True or false, regarding of the status of the condition.

35. How do you link two conditional statements (i.e., a **compound** statement)?

    By using a conjunction (**AND**) or a disjunction (**OR**). [Also called **logical operands.**]

36. How do you create the **truth-table** for compound conditional statements?

    By making a list of all possible outcomes to all combinations of truth-value on the individual statements.

37. What is the purpose of a **loop**?

    Repeat a set of steps to prevent retyping the same instructions over and over, to perform repetitive boring activities, or to perform millions of computations with total precision ("number crunching").

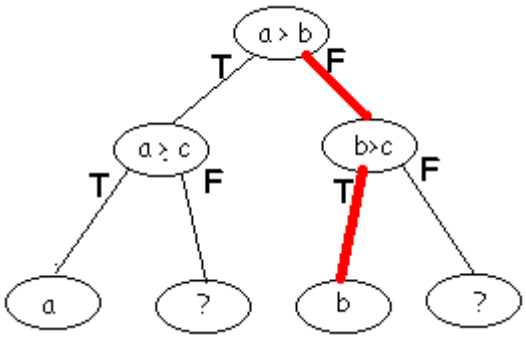
38. What is know as **primitives** of a computing device?

    Basic operations (tasks or instructions) the device is able to follow.

39. Give a **detailed description** to compute the perimeter of a square. (Hint: Remember the quadractic equation example.)

```
Begin
    Request the length of one side and store it in variable L.
    Compute 4*L and store the result in P.
    Return the value stored in P.
End.
```

40. Given the following **decision tree**, indicate the **condition** that has to be true in order to reach the true value at the middle **path.**



$(a \leq b)$ AND $(b > c)$

41. For the sake of the argument, suppose that you have traveled back in time and prevented your parents from meeting each other. Write down the paradox that follows as a **sequence of *if-then*** statements

FORM A (simple wording [T.W.])

- (*) If my parents don't meet, then they don't get married.
- If they don't get married, they didn't have me.
- If they didn't have me, then I didn't travel back in time.
- If I didn't travel back in time, then I didn't prevent them from meeting.
- If I didn't prevent them from meeting, then they got married.
- If they got married, then they had me.
- If they had me, then I traveled back in time so they wouldn't meet… (and go back to the *)

FORM B (with logical operands and predicates):

- Let's my parents be Mr.X and Mrs.Y.
- If {I can travel back in time} then {I can go back to the time when Mr. X and Miss Y were young} AND {avoid them from meet each other}.
- If NOT{Mr X and Miss Y meet each other} then NOT {they will be married}
- If NOT{ Mr X and Miss Y get married} then NOT {I am born.}
- I NOT {I am born.} then NOT {I will travel back in time}…