

OBSERVE THAT `list[10]` DOES NOT REFER TO A MEMORY LOCATION IN ARRAY `list`.

WARNING: IT IS POSSIBLE HOWEVER TO READ OR WRITE PAST THE END OF AN ARRAY BY REFERENCING SAY `list[10]`, `list[11]`, ETC. THIS IS NOT A SYNTAX ERROR IN C/C++ BUT CAN CAUSE RUN TIME ERRORS WHICH ARE VERY DIFFICULT TO TRACK DOWN.

AN ARRAY OF CHARS (LIKE WORD ABOVE) IS CALLED A STRING IN C/C++. THERE ARE CERTAIN SPECIAL OPERATIONS FOR PROCESSING STRINGS IN THE LIBRARY `string.h`, WHICH WE WILL NOT COVER.

C/C++ ALLOWS MULTIDIMENSIONAL ARRAYS

EX `double table[5][6]`

DECLARES AN ARRAY OF 30 doubles WHICH WE MAY THINK OF AS BEING ARRANGED IN 5 ROWS AND 6 COLUMNS:

`table[0][0], . . . . . , table[0][5]`

⋮

`table[4][0], . . . . . , table[4][5]`

THE ELEMENTS OF AN ARRAY CAN BE MANIPULATED LIKE ORDINARY VARIABLES.

EX     int list[10];  
        list[0] = 5;  
        cin >> list[1] >> list[2];  
        cout << list[0];  
        !  
        list[7] = 2 \* list[1] - list[9];

AN ARRAY INDEX CAN BE ANY EXPRESSION WHICH EVALUATES AS AN INT

EX     int i, j, k; int list[10];  
        !  
        list[i+j+k] = -4;

ARRAY ELEMENTS CAN BE INITIALIZED WHEN DECLARED.

EX.     double weights[3] = {2.7, 3.4, 5.1};

DATA AREA

	[0]	[1]	[2]
weights	2.7	3.4	5.1

WHEN INITIALIZING AN ARRAY IN THIS WAY THE DIMENSION IS OPTIONAL.

EX. `double weights[] = {2.7, 3.4, 5.1};`

DOES THE SAME THING.

YOU CAN ALSO INITIALIZE PART OF AN ARRAY THIS WAY.

EX. `double weights[5] = {2.7, 3.4, 5.1};`

DATA AREA:

	[0]	[1]	[2]	[3]	[4]
weights	2.7	3.4	5.0	?	?

CHARACTER ARRAYS CAN ALSO BE INITIALIZED IN THIS WAY.

EX. `char word[] = {'j', 'o', 'y'};`

	[0]	[1]	[2]
word	j	o	y

OR EVEN SHORTER

EX. `char word[] = "joy";`

	[0]	[1]	[2]	[3]
word	j	o	y	\0

\0 IS A SPECIAL CHARACTER CALLED THE NULL CHARACTER, WHICH IS USED AS

A SENTINEL BY THE STRING HANDLING FUNCTIONS IN STRING.H .

## CONTROL STATEMENTS

RECALL THAT THERE ARE THREE TYPES OF CONTROL STRUCTURES .

- 1.) SEQUENTIAL (DEFAULT)
- 2.) CONDITIONAL : DECIDE WHICH INSTRUCTIONS TO EXECUTE BASED ON THE VALUE OF SOME BOOLEAN EXPRESSION .
- 3.) LOOPING : EXECUTE A GROUP OF INSTRUCTIONS MANY TIMES .

RECALL A BOOLEAN EXPRESSION (OR LOGICAL EXPRESSION) IS ONE WHICH CAN BE ASSIGNED ONE OF THE BOOLEAN VALUES true OR false .

Ex.       $a == 0$   
              $b > (c+d)$

BOOLEAN EXPRESSIONS GENERALLY CONTAIN ONE OF C/C++ COMPARISON OPERATOR .

<u>SYMBOL</u>	<u>EXAMPLE</u>	<u>VALUE</u>
==	1 == 2	false
<	1 < 2	true
<=	1 <= 2	true
>	1 > 2	false
>=	1 >= 2	false
!=	1 != 2	true

Ex int a=5, b=6, c=7, d=8;  
THE EXPRESSION

$$(a+b) != (c+d)$$

EVALUATES TO true.

WE CAN BUILD UP LOGICAL EXPRESSIONS BY USING THE C/C++ LOGICAL OPERATORS.

<u>OPERATOR</u>	<u>SYMBOL</u>	<u>EXAMPLE</u>	<u>VALUE</u>
AND	&&	(1 < 2) && (3 > 4)	false
OR		(1 < 2)    (3 > 4)	true
NOT	!	!(1 == 2)	true

NOTE THAT !(1 == 2) IS EQUIVALENT TO 1 != 2.

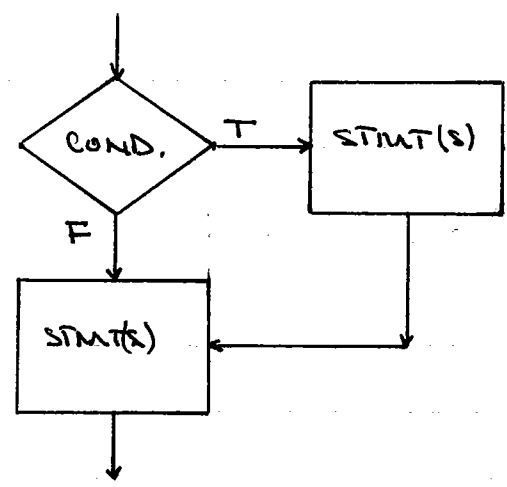
CONDITIONAL EXECUTION: if & if-else

```
if (condition)
    stmt;
```

OR

```
if (condition)
{
    stmt1;
    stmt2;
    :
    stmtk;
}
```

Flow chart



```
EX    if (a == b)
        c = d;
```

```
EX    if (n != 0)
{
    cout << "n is not zero\n";
    c = d;
}
```

AN IF STATEMENT CAN ALSO HAVE AN ELSE CLAUSE.

```

if (condition)
    stmt1;
else
    stmt2;

```

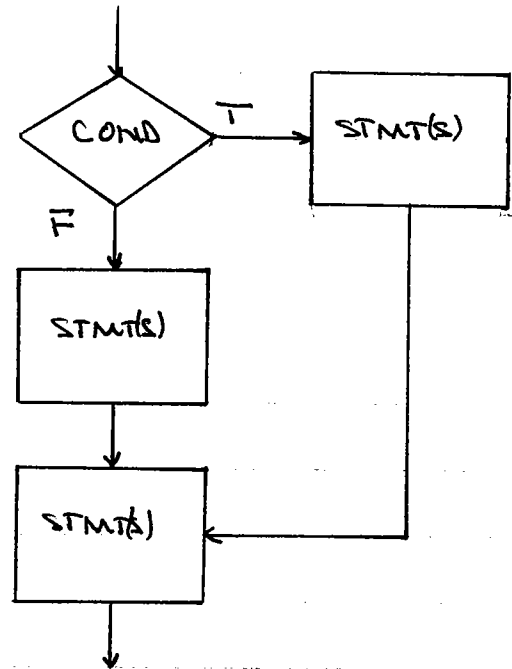
OR

```

if (condition)
{
    stmt1;
    :
    stmtk;
}
else
{
    stmt(k+1);
    !
    stmtn;
}

```

Flow chart



A SEQUENCE OF STATEMENTS ENCLOSED IN BRACES {...} IS CALLED A COMPOUND STATEMENT OR JUST A BLOCK.

EX.

```

if (a == b)
    c = a + 1;
else
    c = b + 1;

```

A CLASSIC ERROR IS TO USE  $a=b$  INSTEAD OF  $a==b$  AS THE CONDITION IN AN `if` OR `if-else` STATEMENT. WHAT HAPPENS IN THIS CASE?

### THE ASSIGNMENT STATEMENT

$$\text{variable} = \text{expression};$$

IS ITSELF AN EXPRESSION WHICH CAN BE EVALUATED. ITS VALUE IS THE VALUE ASSIGNED TO `variable`.

EX  $a = (b = c);$

IS EQUIVALENT TO

$$b = c;$$

$$a = b;$$

FURTHERMORE IF A NUMERIC VALUE IS SUPPLIED WHERE A `bool` IS EXPECTED (SUCH AS THE CONDITION IN AN `if` STATEMENT), THEN NON-ZERO VALUES ARE CONVERTED TO `true`, AND ZERO IS CONVERTED TO `false`.

COMPARE THE FOLLOWING EXAMPLES AND ASSUME THE DECLARATIONS

$$\text{int } a=1, b=2, c;$$