

1 to 10 nos. print.
but you don't want to print
7, so you don't want to
print odd nos.

1 to 100 print
you don't want to print
multiples of 5 & 7, but you
want to print a no. which is
multiple of both

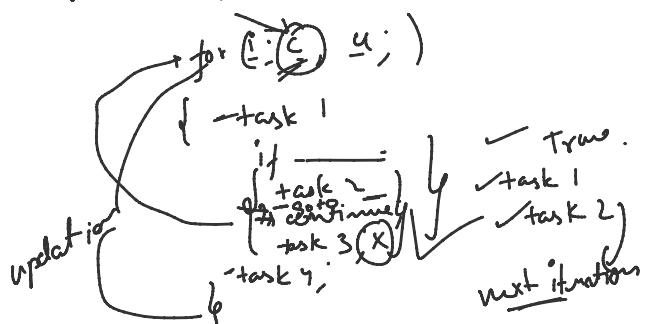
Q. $\times \begin{pmatrix} 5, 7, 10, 14, 15, 21, 25, 28 \\ 30, \end{pmatrix}$

You are given a list of nos.
and you have to find out a specific
no. in that list.

Q. $1, 3, 9, 7, 21$
Number is present
stop looking.

'continue':- It is used when we
want to skip a particular iteration
of a loop.

OR
when code/ compiler encounters continue
statement, it skips the rest of the
code below that statement in that
particular loop only and the execution
of the code goes to the conditions
check for the next iteration
of the loop.



func

WOP.

flow:
point

for (i=1; i<=5; i++):
{ cout << "Hello" << endl;
} Hello will be printed once.

break; It is used to skip all iterations of a particular loop after a particular iteration.

DI.

when wdl/wmbln encounters break; statement, it skips rest of the code below that break statement in that loop and also comes out of that wdl.

for (int i=0; i<5; i++)
{ if (i==3)
 break;
 cout << "Hello" << endl;
}

i = 0 ✓ H - -
 1 ✓ H - -
 2 - H - -
 3
④ X

for ()

for ()

It will be nested
for this wdl only.
break; / continue;

break;

Ans 101, 12 → 1100.

Ans

$$13 \rightarrow 1101, 12 \rightarrow 1100.$$

Given a number, you have to invert the first bit that is set, from right side.

$$^o 1/P \quad 1101 \rightarrow 1100 \rightarrow 12 \text{ o/p.}$$

$(13) \rightarrow (12)$

$$^o 1/P_{12} \rightarrow 1100 \rightarrow 1000 \rightarrow 0 \text{ o/p.}$$

$8 \rightarrow 0$

$$17 = 10001 \rightarrow 16$$

$\{ 'f', 't', 'z', 'n', '<', '>' \}$

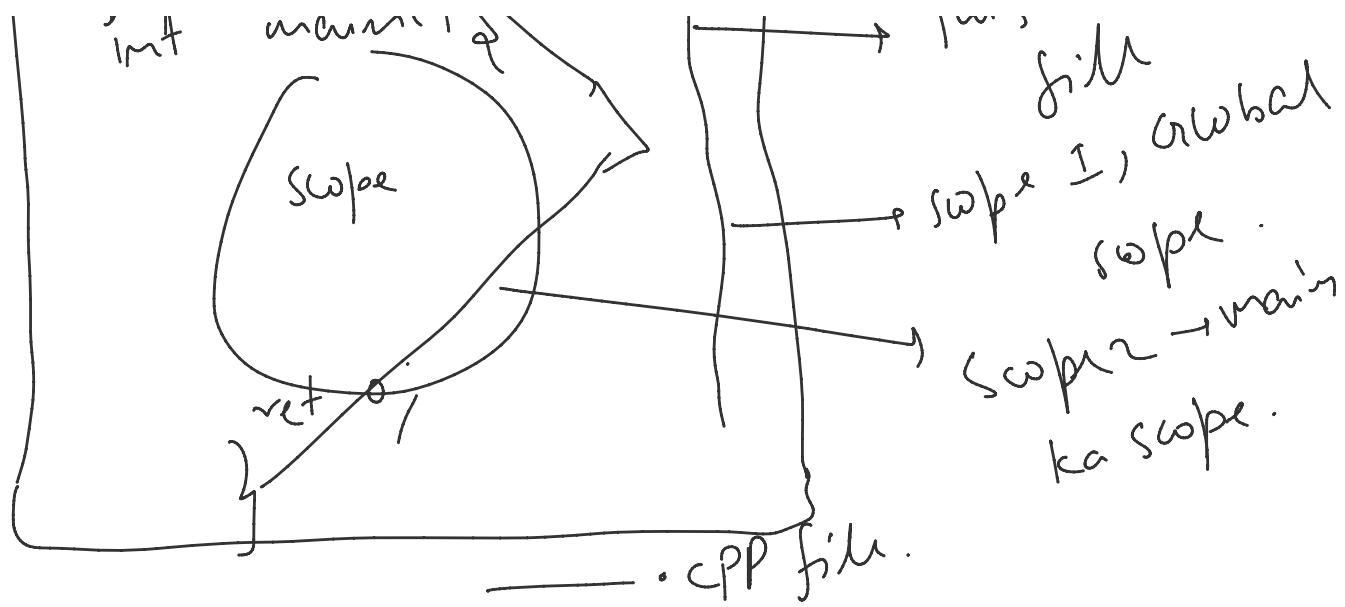
India
E =

USA
\$ X

Scope is defined by curly braces.

int main() { } This much file .. in

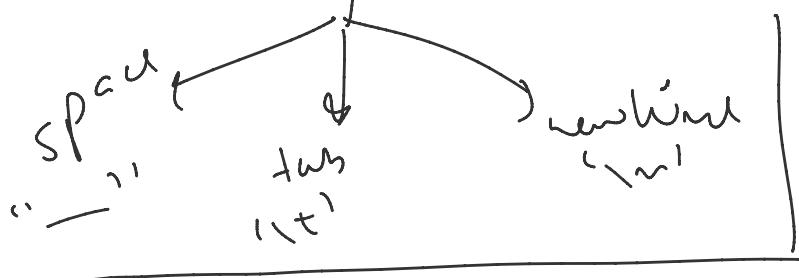
Diagram: A curly brace is shown under the code block 'int main() { }'. An arrow points from the brace to the text 'Scope is defined by curly braces.'. Another arrow points from the brace to the text 'This much file .. in'.



you have to give input to a program:

~~helloworld~~ → count = (10) .
~~helloworld~~ → (12)

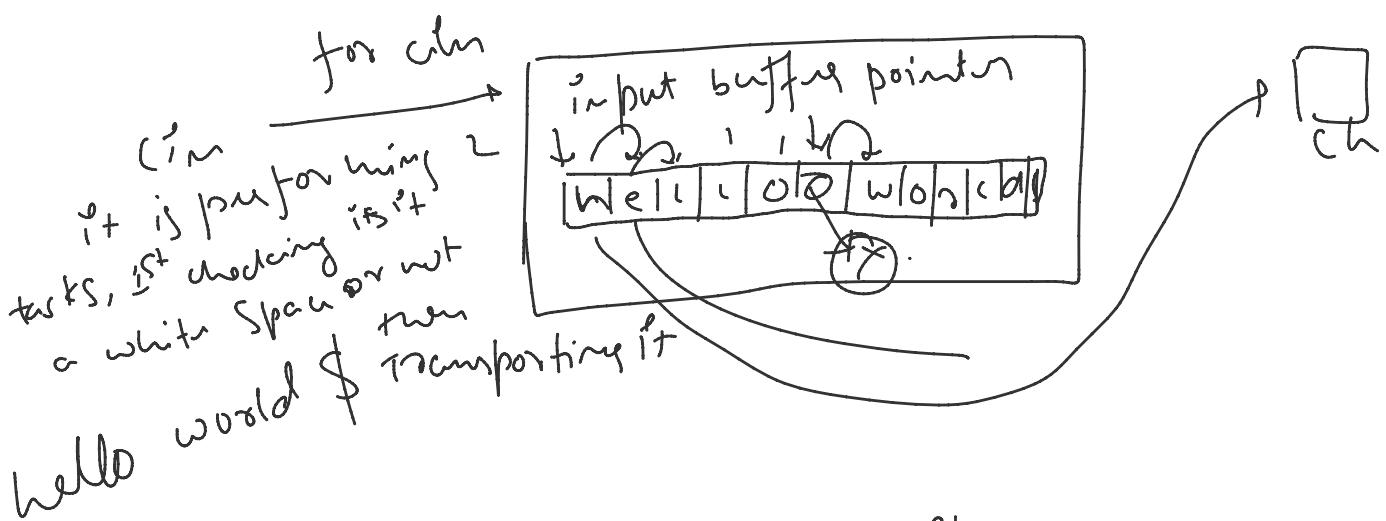
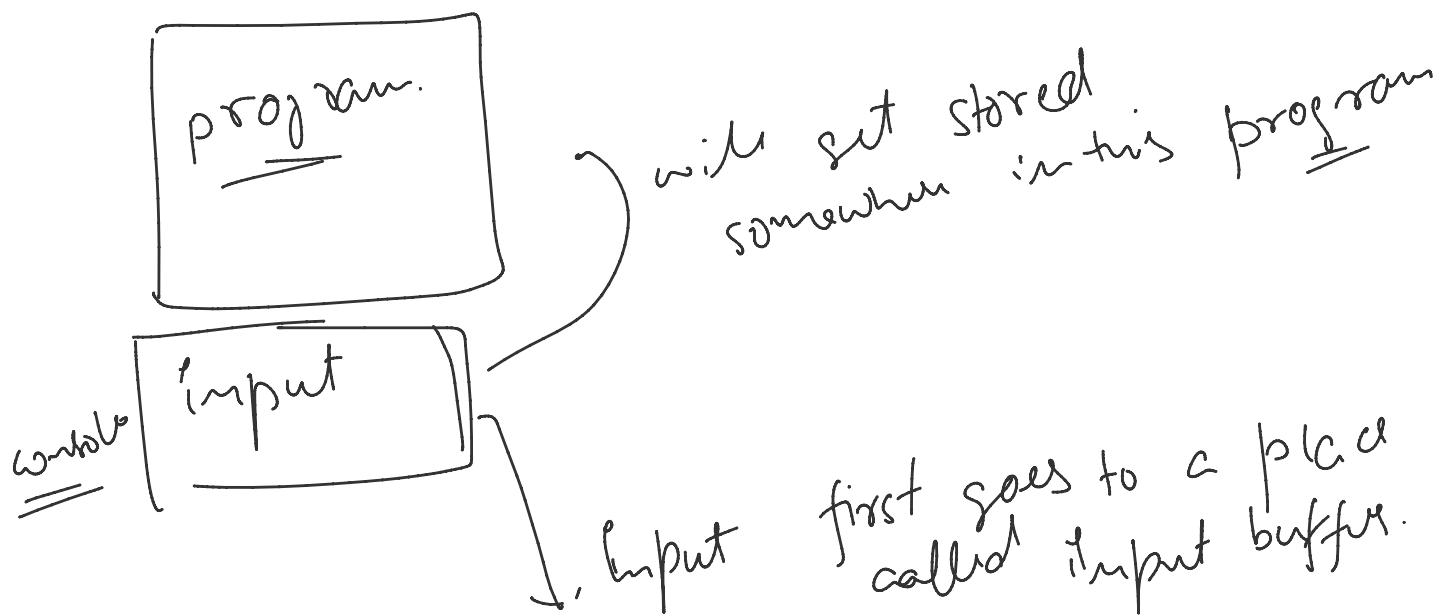
→ whitespace is also a character
 but cin will ignore it.



cin vs cin.get()

`cin` vs `cin.get()`

when you are using `cin`, it has a property to ignore white space.



`cin.get()` will not skip.
only single task \rightarrow transport