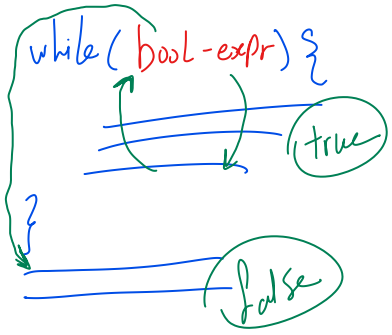

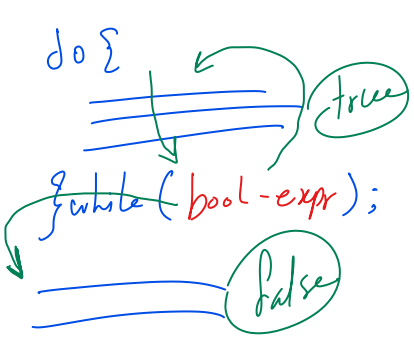

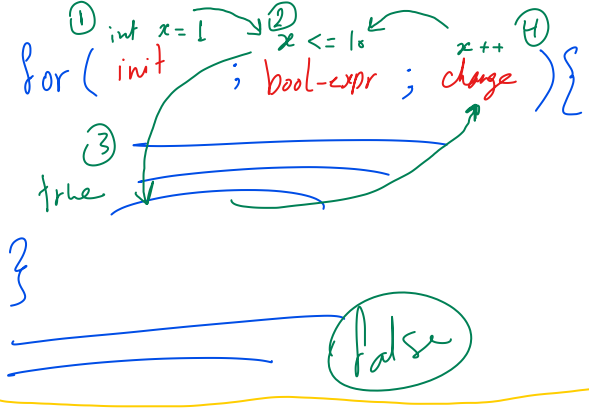

what's 5 * 2 ?
» 10
what's 7 * 6 ?
» 42
what's 3 * 7 ?
» ⋮
» 35
Game over

what's 5 * 2 ?
» 15
Try again, what's 5 * 2 ?
» 30
Try again, what's 5 * 2 ?
» ⋮
» 10
Correct

Guess num btw 0-100
» 50
Go down
» 40
Go down
» 25
Go up
» 30
Go down
» 29
You Guessed Right!

number
29

guess
~~50~~
~~40~~
~~25~~
~~30~~
29

what's 3 * 2 ?
» 6
Correct
what's 7 * 4 ?
» 28
Correct
what's 6 * 9 ?
» 70
Wrong
———
—
———
———
—
———
Your score is 3 out of 5

Enter num of std.: 4
Enter exam full mark: 25
Enter std mark: 23
pct = mark / fullMark * 100
>=50 Pass
Fail

Print first 50 prime numbers

number: 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, ...

Print: 2, 3, 5, 7, 11

all cannot divide → prime → print

one divide → break → not prime → no Print

while _____:

for _____:

if _____:

break

x, y

8, 24

gcd	div
~~1~~	2
~~2~~	~~3~~
~~4~~	~~4~~
8	~~5~~
	~~6~~
	~~7~~
	8

string x = "Hello";

cw(x); → Hello

cw(x[0]); → H

cw(x.Length); → 5

x

H	e	l	l	o
0	1	2	3	4

x.

char.

Ahmed123 ✗

20Ahmed20 ✓

① at least 8 char

② at least 2 digits

③ at least 2 upper

digitCount
~~0~~
~~1~~
~~2~~
~~3~~
4

upperCount
~~0~~
~~1~~
2

05XXXXXXXX

① length 10 char

② starts with 05

③ all digits

digitCount
~~0~~
~~1~~
~~2~~
~~3~~
~~4~~
~~5~~
~~6~~
~~7~~
~~8~~
~~9~~
10

reuse, organize, debug

Parameters → MethodName → Code → return

Define Method

modifiers return-type MethodName(Parameters) {

}

Call Method

Math.Min(5,2) → 2

Math.Pow(5,2) → 25

Math.Round(-)

char.IsDigit(-)

weight, height

$bmi = \frac{weight}{(height/100)^2}$ → GetBMI(w,h)

< 18.5 Underweight

< 25 Normal

< 30 Overweight

Obese

→ GetStatus(bmi)

mark, fullMark

$pct = \frac{mark}{fullMark} \times 100$ → GetPercent(-,-)

>85 Excellent

>75 V. Good

>65 Good

>=50 Pass

Fail

→ GetGrade(pct)