Sum of digits of a number → 462 → 12

Number

ii) 462 →

4 6 2

462 % 10

1 10 ←

4 | 6 | 2

$4^2 + 6^2 + 2^2$

12 + { 4 → 1 digit
6 → -
2 → - }

462 → 2

462 % 10

46 / 10

4

$n = 462$ → $4 + 6 + 2$

(3) (2) (1)

$2 + 6 + 4$ → 12

Dry run

Steps

(1) $n = 462 \% 10$ → 2 — (1)

(2) $n = 462 / 10$ → 46

(3) $46 \% 10$ → 6 — (2)

(4) $46 / 10$ → 4 — (3)

46

$n = 462$

Loops

Assignment-1

- Sum of digits of a number
- Sum of square of digits of a number
- multiplication of 4 digits number-1234=24
- Reverse of a number without using loop
- Print the total number of days message in month of Year
 for eg: month =3
 march month has 31 days
- Check number is even or odd

Conditional operators

→ <, >, <=, >=, ==, !=

We can use with if..else statements.

Conditional Statements / Control Structure

→ Decision making statements

- → if
- → ifelse
- → if elseif else → Ladder elseif
- → Nested if else

→ Switch case stmt

(ii) Loop Statement

→ for,

→ while

→ dowhile

→ foreach → only for Arrays.

(iii) jumping → goto ✗

Continue;

break;

(i) Decision Making Stmts

→ if → keyword

Syntax

```
if ( expression ) {
    stmt → execute this
}
```

condition or

expression → true

3

eg. 10 > 5 → true

true / false

```
if ( 10 > 50 ) {
    Syso ("Yes");
}
```

Check whether a Number is even or odd.

```
int n = 40;

rem = n % 2  →  0

if ( rem == 0)
{
    Syso("even");
}else{
    syso("odd");
}
```

→ 40 / 2

2 | 40 | 20

40

0 → even

16

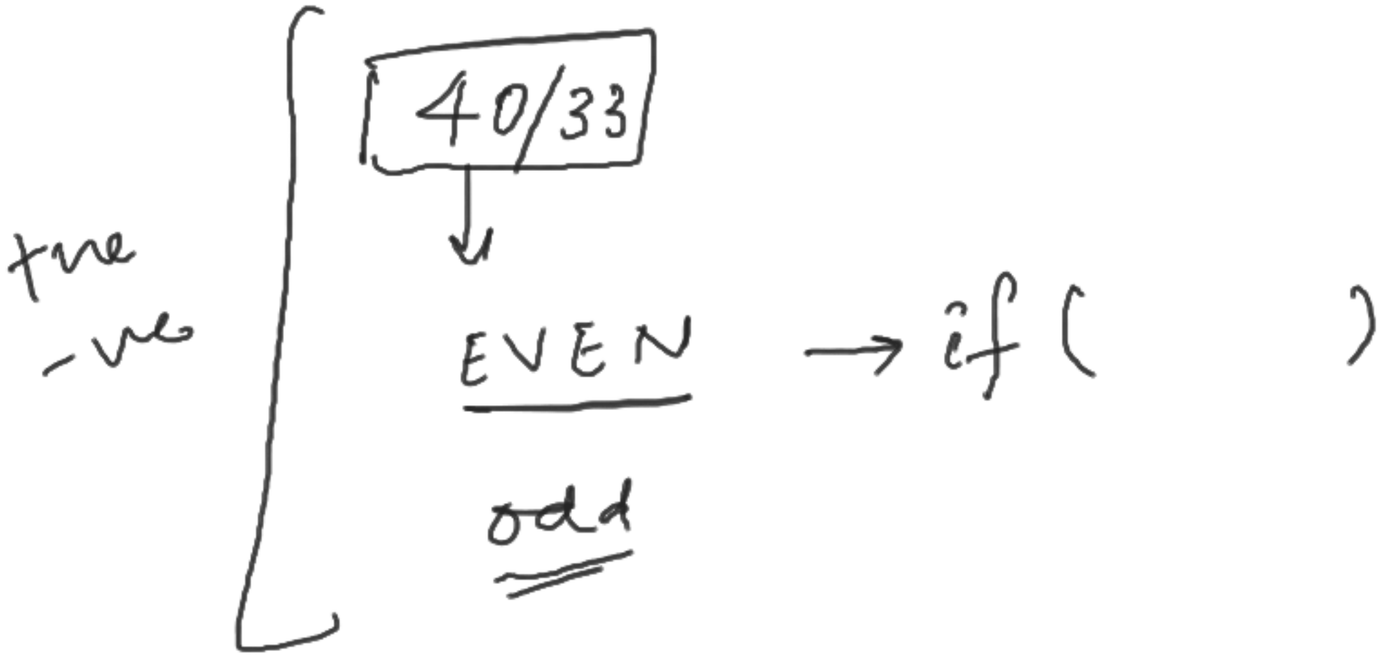
2 | 33 → odd

2

13

12

1 → x odd


40/33
+ve
-ve
EVEN
odd
→ if ( )

if-elseif-else

true | false

Syntax :-

```
if (exp) {
    //s1
} else if (exp) {
    //s2
} else {
    //s3
}
```

(1) x

false → (exp) → true → (2) only

for multiple conditions

Logical operators with control stmts

`&&` → AND

`||` → OR

`!` → NOT

I/P1	I/P2	→ O/P
true	true	→ true ✓
true	false	→ false
false	true	→ false
false	false	→ false

Ex: username && password

V && V → V

V I → I

I V → I

I I → I

OR `||`	I/P1	I/P2	→ O/P
	true	true	→ true
	true	false	→ true
	false	true	→ true
	false	false	→ false

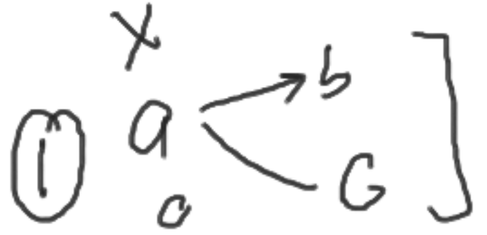
Find largest No. out of three numbers :-

a = 45, b = 23, c = 100

(i) a > b && a > c

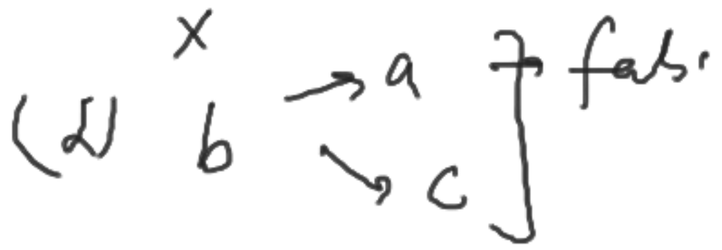
yes No

true false → false



(ii) (b > a) && (b > c)

false false → false



(iii) else { c is largest }

Example of OR (||) operator

→

(username || email) → 2 ✓

AND

(password || DOB) →

(1 == 1) ✓

(1 == 1) ✗

("Test" == "Test") → String methods

To compare strings we can't use (==) operator.

55 22

↓ 5 11

→ only 5

→ only 11 ✓

→ Both

→ none of both

WAP to print the grades acc to percentage range.

marks = 65

→ — <=50 → Average

✓ 51 - 70 = Good

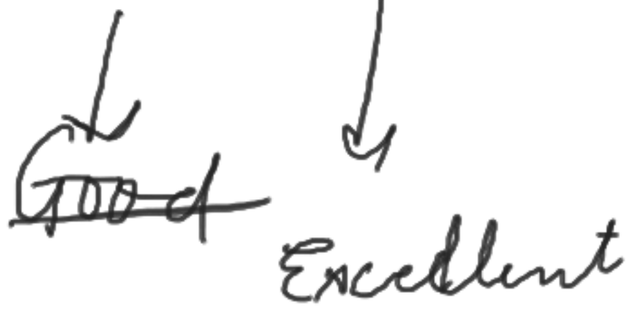
71 - 85 → V. Good

> 85 → Excellent

90

marks → ~~65~~

55




```
if (marks <= 50)
  {
    Syso ("Average");
  }else if ((marks >= 51) && (marks <= 70))
```

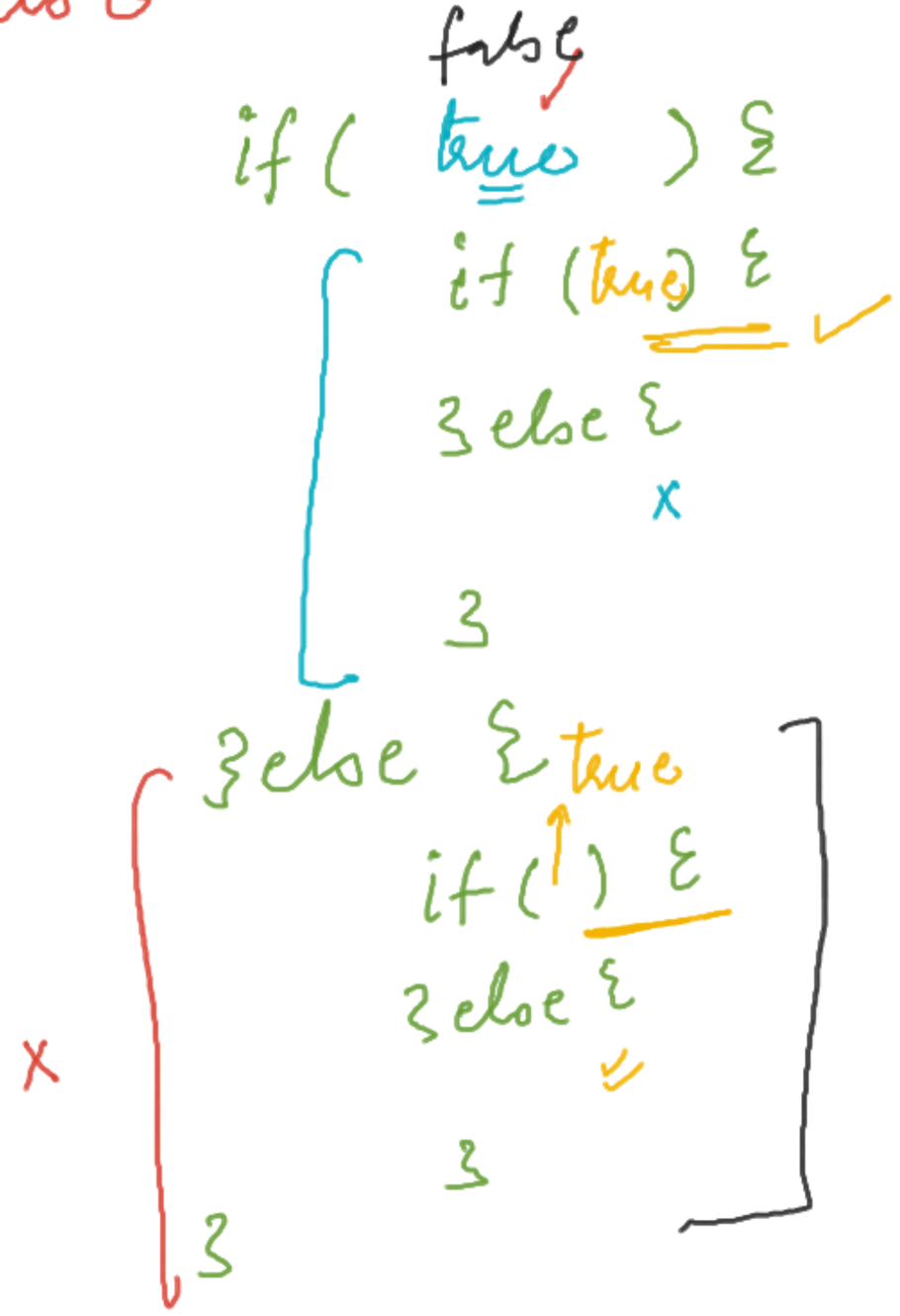
80] true ✓✓ — false

100

60 <= 70

Nested if else

Syntax



Largest No. out of 3 using Nested if else.

```
if( a > b )          {
    if(a > c)        {
        Syso("a is Largest");
    }else{
        Syso("c is Largest");
    }
}else{
}
```

true

a = 110
b = 20
c = 4

110 > 20 → true
110 > 4

Largest

3

int month = 4 ; — April

0 ✓, 13

→ April month has 30 days.

year 12 → Dech — 31 —

12

```
if (months==1) {
}else if (months==2) {
}else if ( ) {
}else if ( ) {
}else {
}
```

1 — 12

number is divisible by 5 and 11

int num = 22;

if ((num % 5 == 0) && (num % 11 == 0)) {

22 % 5 → false

true

false

} else {

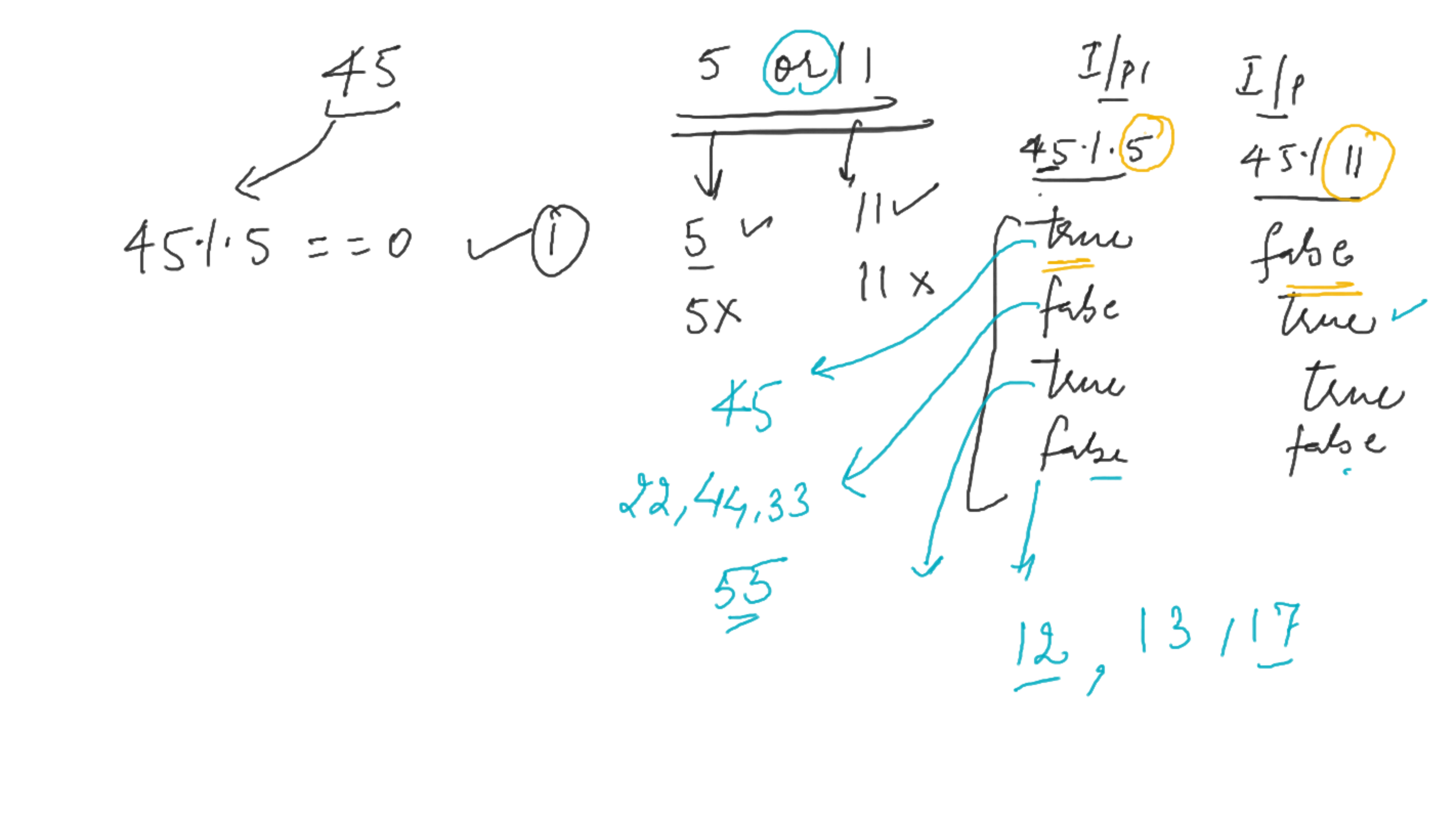
}

3

5 | 22 → 4, 20, 2 ✗

% == 0 → divisible

11 | 22 → 2, 22, 0


45
45·1·5 == 0
1
5 or 11
5
11
5X
11 X
I/p1
45·1·5
I/p
45·1 11
true
false
false
true
true
false
45
22, 44, 33
55
12, 13, 17