



Daughter's Ages

$$x \ y \ z = 36$$

$$x + y + z = \underline{\text{House No}}$$

$$1, 1, 36 \rightarrow 39$$

$$1, 2, 18 \rightarrow 21$$

$$1, 3, 12 \rightarrow 16$$

$$1, 4, 9 \rightarrow 14$$

$$1, 6, 6 \rightarrow 13$$

$$2, 2, 9 \rightarrow 13$$

$$\rightarrow 2, 3, 6 \rightarrow 11$$

$$\rightarrow 3, 3, 4 \rightarrow 10$$

~~7, 1~~

Confusions

$$\begin{aligned} x \ y \ z &= 36 \\ x + y + z &= 13 \end{aligned}$$

2, 2, 9

correct :-)

Branching

[if
if - else
if - elseif - else

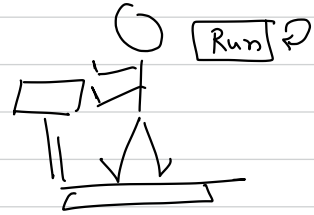
Loops

{ while loop
for loop
do while

Task → Repeated

Cal → 100

Cal = 0 → init val
while (Cal < 100) {
 Run 1 step
 Cal = Cal + 1
}
Come here Workout is Complete



Cal = Cal + 5

cal = 0

while cal < 100 {

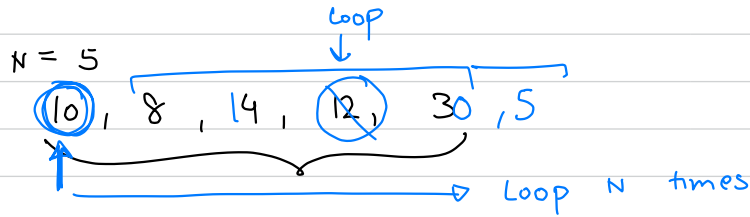
cal = cal + 5

}

Q

Take Input N, followed by N numbers.
Find out the largest NO

L = ∞



Largest Till Now



NO

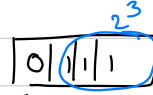
Doubles

2's



-2^3

to $2^3 - 1$



↑ 4 bits

Sign bit

-8

8 No

to 0 to 7

8 No

= 16 Nos

-2^3 to $2^3 - 1$



$2^2 \ 2^2 \ 2^2 \ 2^2$
x x x x

2^4 combinations = 16 combinations

8 Binary

Sign bit +ve

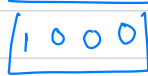


$-(-x)$



↙

+ 0 1 1 1

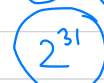


↑
magnitude ⇒ 8

$-2^{31} - 1$ to $0 - 2^{31} - 1$



-ve



+ve

2^{32} unique No

-20000

largest

-40

-20

Largest ⇒

-10

BigInteger

~~Arrays~~

-40

↑
val

-50

↑
val

-20

↑
val

-10

↑
val

-15

↑
val

No

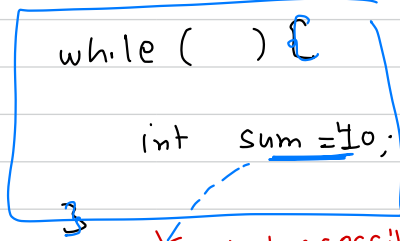
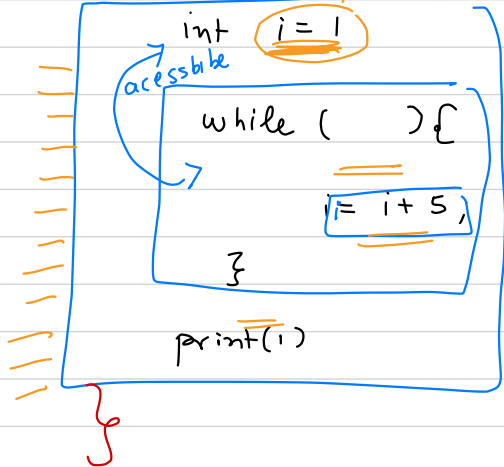
Scope

{ ... }

Block Scope

class Demo {

= main() {



~~Not accessible~~

}

Handwritten Java code snippet with annotations:

```

public class Hello
    = Scanner SC
{
    main() {
        i = 1
        Loop()
        sam
    }
    3
    sayHi () {
        x = 5
    }
    3
}
  
```

Annotations:

- A blue arrow points to the `class` keyword.
- A blue box highlights the `main()` method signature.
- A red double-headed arrow indicates a relationship between `main()` and the `sayHi()` method.
- A blue double-headed arrow indicates a relationship between `Loop()` and `sam`.
- A long orange arrow points from the `main()` method down to the `sayHi()` method.
- A blue arrow points from the `sayHi()` method down to the closing curly brace of the class.

class

For Loop

↓

compact syntax (easy)

```
for (int cal = 0; cal <= 10; cal = cal + 1) {  
    Run ---- ;  
}
```

<=>

while

```
int cal = 0;  
while (cal <= 10) {  
    Run ----  
    cal = cal + 1;  
}
```

Operator

++

Post
inc

Pre
inc

x = 10

y = x++

--

Pre
dec

Post
dec

x = 10

y = --x

$y++$

$x = x + 1$

Let $x = 10$

$y = x++$

Post inc

$\rightarrow y = x$
 $x = x + 1$

10
y

11
x

12
x

12

$y = ++x$

Pre inc

$x = x + 1$

$y = x$

Compound Assignment Operators

$a = a + 5$ ✓ Same

$a = a * b$

$a = a - 10$

$a = a \% b$

$a = 10$

5

Shortcut ✓

$a += 5$

$a *= b$

~~$a =$~~

$a -= 10$

$a /= b$

$a /= 2$

$a = 5$

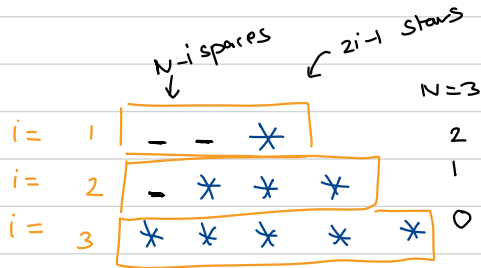
$b = 2$

$a \% b$

$\Rightarrow a = 1$

Break Time

10 30



for ($i=1$; $i \leq N$; $i=i+1$) {

for (spaces = 1 ; spaces \leq $N-i$; spaces++)
 print(" ")

3

for (stars = 1 , stars \leq $2i-1$; stars++)
 print "*"

print()

3

}

}

1 \rightarrow 1

2 \rightarrow 3

3 \rightarrow 5



one less than $\frac{2i}{2}$
 $2i-1$

$N=4$

$i=1$

1

$i=2$

1 2

$i=3$

1 2 3

$i=4$

1 2 3 4

col \rightarrow to i

val

$i=1$

1

val is not same as col

$i=2$

2 3

$i=3$

4 5 6

$i=4$

7 8 9 10

$N=4$

000010000
000232000
003454300
045676540
567898765

$i=1$ 1
 $i=2$ 2 3 2
 $i=3$ 3 4 5 4 3
 $i=4$ 4 5 6 7 6 5 4

$N-i$ spaces

i No's \uparrow val $\rightarrow i$
 $i-1$ No's \downarrow

HW

val
start of
prev
+ i

1
32
654
10987

Break & make

Start
i=2

i=1

①
3
6
10

start 0
start p+1 = 1
1+2=3
3+3=6
6+4=10

i=1 A
i=2 A B
i=3 A B C
i=4 A B C D

char val = 'A' - 65
B - 66

char val = val + 1
print(val)

fs
6
B



\Rightarrow

