

$s = \text{CCATGHAATAADEFFA}$
 start ↓ stop ↓
 0 1 2 3 4 5 6 7 8 9 10

gene =
 $s.\text{substring}(\text{start}, \text{stop} + 3)$

$\text{start} = 2$
 $\text{stop} = 8$

$\text{stop} + (2 \times \text{start}) - 3$

$\text{len} = (\text{stop} - \text{start}) + 3$

$s.\text{substring}(3, 7) \rightarrow \text{DEFG}$
 ↓ 1 1 1 1
 ABCDEFGHI
 0 1 2 3 4 5 6 7 8

$\text{dna}.\text{substring}(\text{start}, \text{stop} + 3)$

$\text{len} = (\text{stop} + 3) - \text{start}$

If $x+3$ is a multiple of 3
 is x a multiple of 3

```
public String findGene (String dna) {  
    String startCodon = "ATG";  
    String stopCodon = "TAA";
```

```
    ① int startIndex = dna.indexOf(startCodon);  
    if (sI == -1) {  
        return "";  
    }
```

```
    ② int currIndex = dna.indexOf(stopCodon, sI + 3);
```

```
    ③ while (cI != -1) {  
        int len = cI - sI + 3;
```

```
        ④ if (len % 3 == 0) {
```

```
            ⑤ String gene = dna.substring(sI, cI + 3);  
            return gene;  
        }
```

```
        ⑥ cI = dna.indexOf(stopCodon, cI + 3);  
    }
```

```
    ⑦ return "";
```

```
}
```

```
boolean isVowel (char c) {  
    ...  
}
```

```
for (int i=0; i<s.length(); i++){  
    char c = s.charAt(i);  
    if (!isVowel(c)) {  
        // if c is not a vowel  
    } else {  
        // if c is a vowel  
    }  
}
```

```
if (isVowel(c) == true)  
    ↓  
if (isVowel(c))
```

```
if (isVowel(c) == false)  
    ↓  
if (!isVowel(c))
```

["CA", "Illinois", "MA",
"Ohio", "Nevada", "NC", "Kansas"]

⁰
[CA, ¹MA, ²NC] $n_a = 3$

[Illinois, Ohio, Nevada, Kansas] $n_b = 4$

0 : CA

4 : Nevada

1 : Illinois

5 : NC

2 : MA

6 : Kansas

3 : Ohio

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
 H S A T G A T C G C T G A T T A G G C T T A A A T G A C G
 ↑

1. find ATG
 start = 2

2. taaIndex = findStopCodon(dna, start, "TAA")

↓
 20

3. tagIndex = findStopCodon(dna, start, "TAG")

↓
 14

4. tgaIndex = findStopCodon(dna, start, "TGA")

↓
 -1

5. minIndex = 14

6. gene = dna.substring(start, minIndex + 3)

taaIndex,
 tagIndex
 tgaIndex

minIndex = +100
 if (taaIndex != -1 && taaIndex < minIndex)
 {
 minIndex = taaIndex
 }

if (tagIndex != -1 && tagIndex < minIndex)
 {
 minIndex = tagIndex;
 }

findStopCodon(dna, start, stopCodon)

↓
returns dna.length() if
a valid stopCodon is
not found

minIndex = taaIndex

if (tgaIndex < minIndex) {
 minIndex = tgaIndex
}

if (tagIndex < minIndex) {
 minIndex = tagIndex
}

if (minIndex == dna.length()) {
 return ""
}

return dna.substring(start,
 minIndex + 3)

String findGene(String dna) {

1. find startCodon & put in
startIndex

2. if done return ""

3. taaIndex,
tgaIndex
tagIndex

2. find minIndex

5. If minIndex is invalid
return

6. return gene

}

int findStopCodon(String dna,
 int start,
 String stopCodon) {

...
}

s = "123 -345 -112 6457"

2

String currNum = "6457" "6457 -345"

h = 123, l = -345

i = 0

ch = '1'

i = 1

ch = '2'

i = 2

ch = '3'

i = 3

ch = ' '

i = 4

ch = '-'

T		T	=	T	T	&&	T	=	T
T		F	=	T	T	&&	F	=	F
F		T	=	T	F	&&	T	=	F
F		F	=	F	F	&&	F	=	F

\uparrow true
 if (condA || condB) {
 print(hello)
 }

In OR, if 'condA' is true,
 there is no need to evaluate
 'condB' as result will be true
 irrespective of whether 'condB'
 evaluates to T/F.

x = 0
 if (53 % x == 0) {
 print("53 is divisible by " + x);
 }

crashes if x is 0

\uparrow false
 if (condA && condB) {
 ...
 }

Java won't evaluate 'condB'
 if 'condA' is false (in && statements)

\uparrow false
 if (x != 0 && 53 % x == 0) {
 print("...")
 }

\rightarrow Error
 (53 % x == 0)

$$\boxed{3 < 2}^F \text{ \& \& } (5 < 10 \text{ \& \& } 6 \leq 6) = F$$

$$\boxed{1 < 2}^T \text{ || } (3 < 5 \text{ || } (5 < 6 \text{ \& \& } 7 > 8)) = T$$

$$\boxed{3 \leq 2}^F \text{ || } (\underbrace{\boxed{5 < 10}^T \text{ || } (5 < 6 \text{ \& \& } 7 > 8))^T = T$$

findGene(dna, start) → returns
gene
string if
present
otherwise
returns ""

```
printAllGenes(dna) {  
  start = 0;  
  while(true) {  
    gene = findGene(dna, start);  
  
    if (gene.isEmpty()) {  
      break;  
    }  
    print(gene);  
  
    start = dna.indexOf(gene, start)  
              + gene.length();  
  }  
}
```

0 2 8 11 17
↓ ↓ ↓ ↓ ↓
L A A T G T A A H I D A T G T A A T T T
gene

0 1 2 3 4 5 6 7 8 9 10

A B C r o h a n A B r o h a n D E

start=0, count=0

while (true) {

 index = s.indexOf("rohan", start)

 if (index == -1) { break; }

 count += 1

 start = index + "rohan".length;

}

$\text{len}-3$ $\text{len}-1$
...abc
 \downarrow
 $\text{len}-2$

$\text{len}-4$
...abc_ \downarrow $\text{len}-1$

\Rightarrow not OK if $\boxed{\text{index} \geq \text{len}-3}$
(of abc)

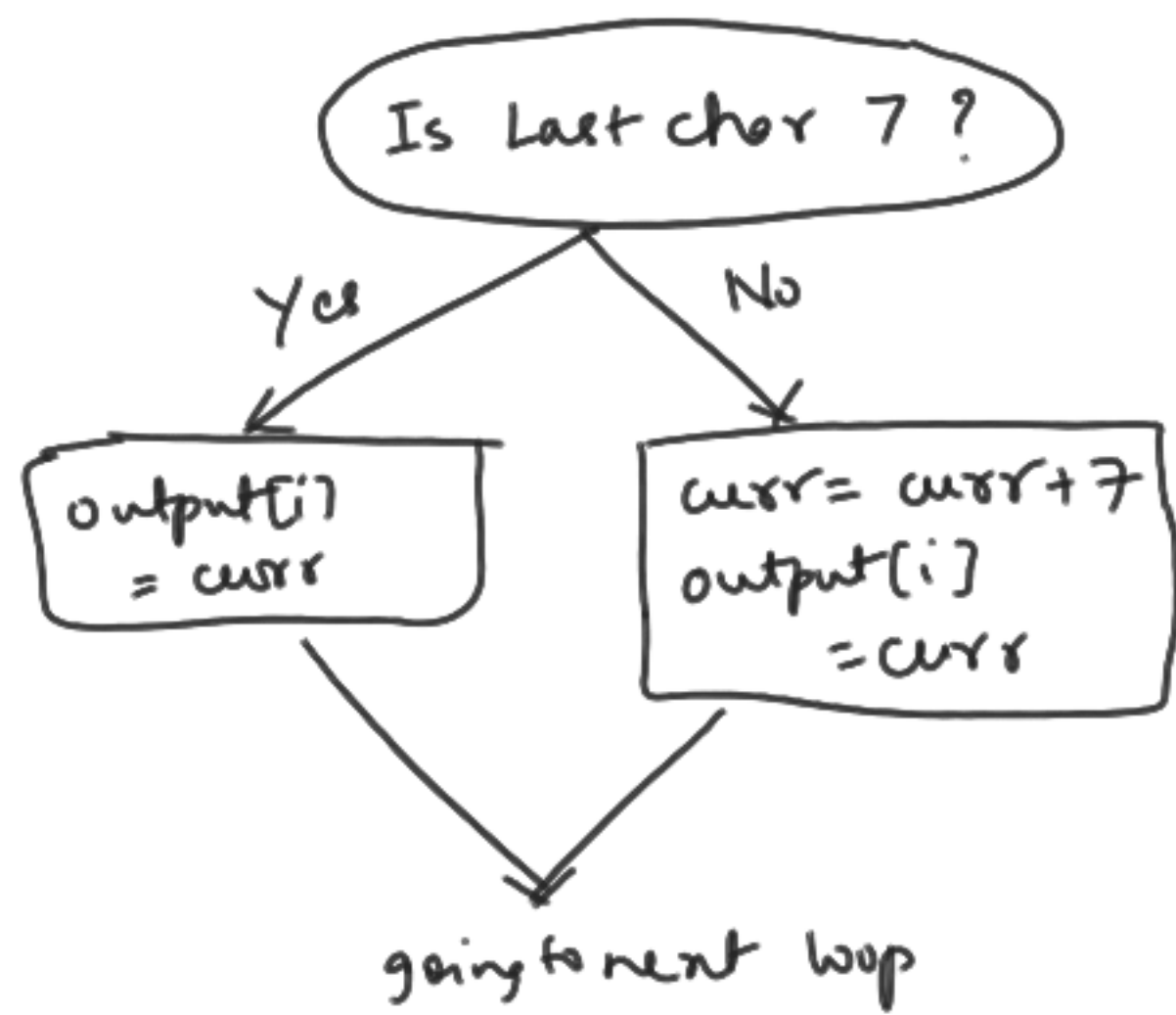
$\boxed{\text{index} > \text{len}-4}$

substring(index+1,
index+4)
 \downarrow
max value
of second param
= length

Invalid
if second param > length

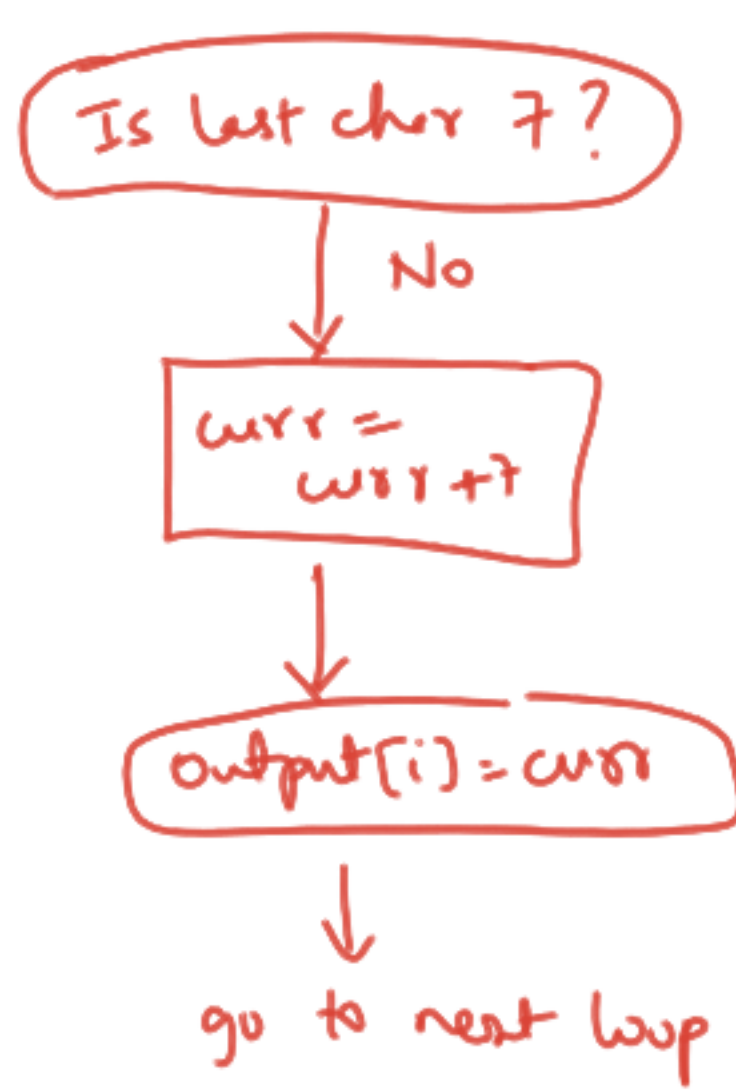
$\Rightarrow \text{index} + 4 > \text{len}$

$\Rightarrow \boxed{\begin{array}{l} \text{index} > \text{len} - 4 \\ \text{index} \geq \text{len} - 3 \end{array}}$



```

if (lastChar == '7') {
    output[i] = curr;
} else {
    curr = curr + '7';
    output[i] = curr;
}
...
  
```



```

if (lastChar != '7') {
    curr = curr + '7';
}
output[i] = curr;
...
  
```

```

void printAllGenes(dna) {
    ...
    while (true) {
        ...
        // when gene is found
        print(gene);
        ...
    }
}

```

```

int countAllGenes(dna) {
    ...
    while (true) {
        ...
        // gene found
        count += 1;
        ...
    }
    return count;
}

```

```

StorageResource getAllGenes(dna) {
    SR list = new SR();
    ...
    while (true) {
        ...
        // gene found
        list.add(gene);
        ...
    }
    return list;
}

```

```

StorageResource gl = getAllGenes(dna);
for (String gene : gl.data()) {
    int len = gene.length();
    if (len > 10) { print(gene); }
}

```

```

geneList
StorageResources = getAllGenes(dna);
for (String gene : geneList.data()) {
    print(gene);
}

```

```

count = 0
SR geneList = getAllGenes(dna);
for (String gene : geneList) {
    count += 1;
}

```


pos = 27
count = 3
start pos 22
newDna

C A B C T ... T ... T ... T ... T ...
0 1 2 3 4 9 21 27

C A B C ...

50 miles \rightarrow 50×1.1^2
increase
the distance
by 10%
twice

$$50 + \left(50 \times \frac{10}{100}\right)$$

$$= 50 \times 1.1$$

$$(50 \times 1.1) \times 1.1$$

$$8.6 \times 1.05^3 \times 1.1$$

$$'x' \text{ ltr}/100 \text{ km}$$

& we have
'y' ltr
of fuel

$$x = 60 \text{ ltr}/100 \text{ km}$$

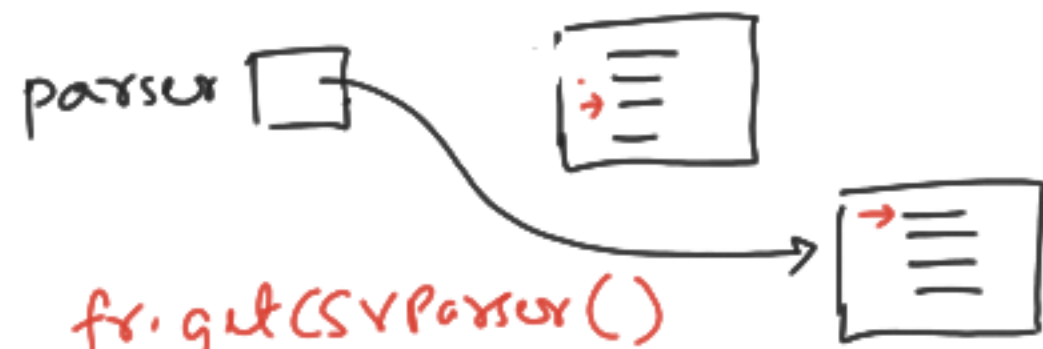
$$y = 120 \text{ ltr}$$

$$x \text{ ltr} - 100 \text{ km}$$

$$y \text{ ltr} - d$$

$$\frac{x}{y} = \frac{100}{d}$$

$$d = \frac{100y}{x}$$



fr. getCSVParser()
countryInfo(parser, country)
listCountries(parser, exp1, exp2)

Temp	Hum	Time
32	4	11am
25	7	12pm
31	3	1pm

ColIndexSoFar = SecondRow

CSVRecord rlt = null

```
for (record : parses) {  
    if (rlt == null) {  
        rlt = record;  
    }  
    else {
```

curTemp = Double.parseDouble(record.get("Temp"));

lowTemp = Double.parseDouble(rlt.get("Temp"));

if (curTemp < lowTemp) {
 rlt = record;

}

}

}

x = 5

if (x < 10) {
 print("hello");

}
 if (x < 20) {
 print("world");
 }

File fIt = null;

double lt = ;

for (file : files) {

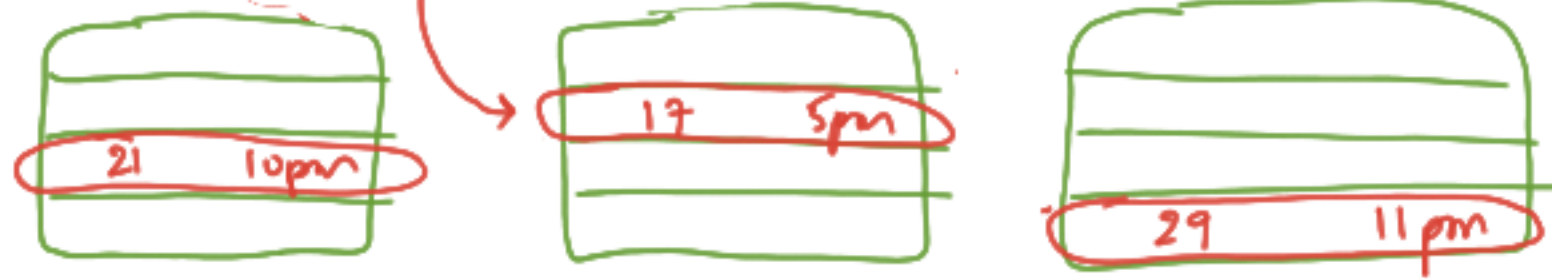
	humidity	time
1	25	8 am
2	27	9 am
3	25	10 am
4	24	11 am
5	26	12 pm
6	24	1 pm
7	24	2 pm
8	32	3 pm

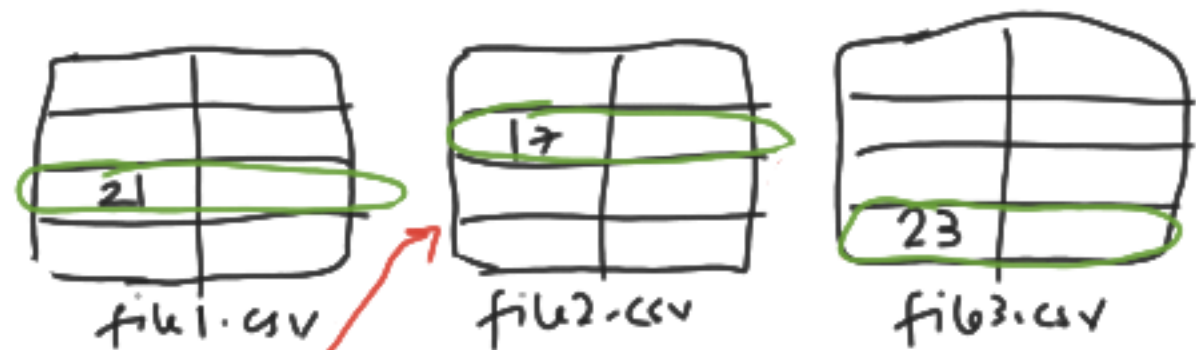
ans if first tie is taken

ans if last tie is taken

LHRAF

curr File
curr File's lowest HumRead





l & r
l = 17
curr
curr Read
curr Temp

Temp	Humidity
10	61
20	79
30	80
40	63

$\frac{(10+30+40)}{3}$

String a = "rohan";
String b = "rohan kumar";
String c = "rohan";
a == c → false
a.equals(c) → true
b.equals(a) → false
a.contains(b) → false
b.contains(a) → true
a.contains(c) → true
b.contains(c) → true
a == a → true
a.equals(a) → true

Emma, F, 25
Rebecca, F, 10
John, M, 30
Peter, M, 7

} Num of male names = 2
Num of females = 3