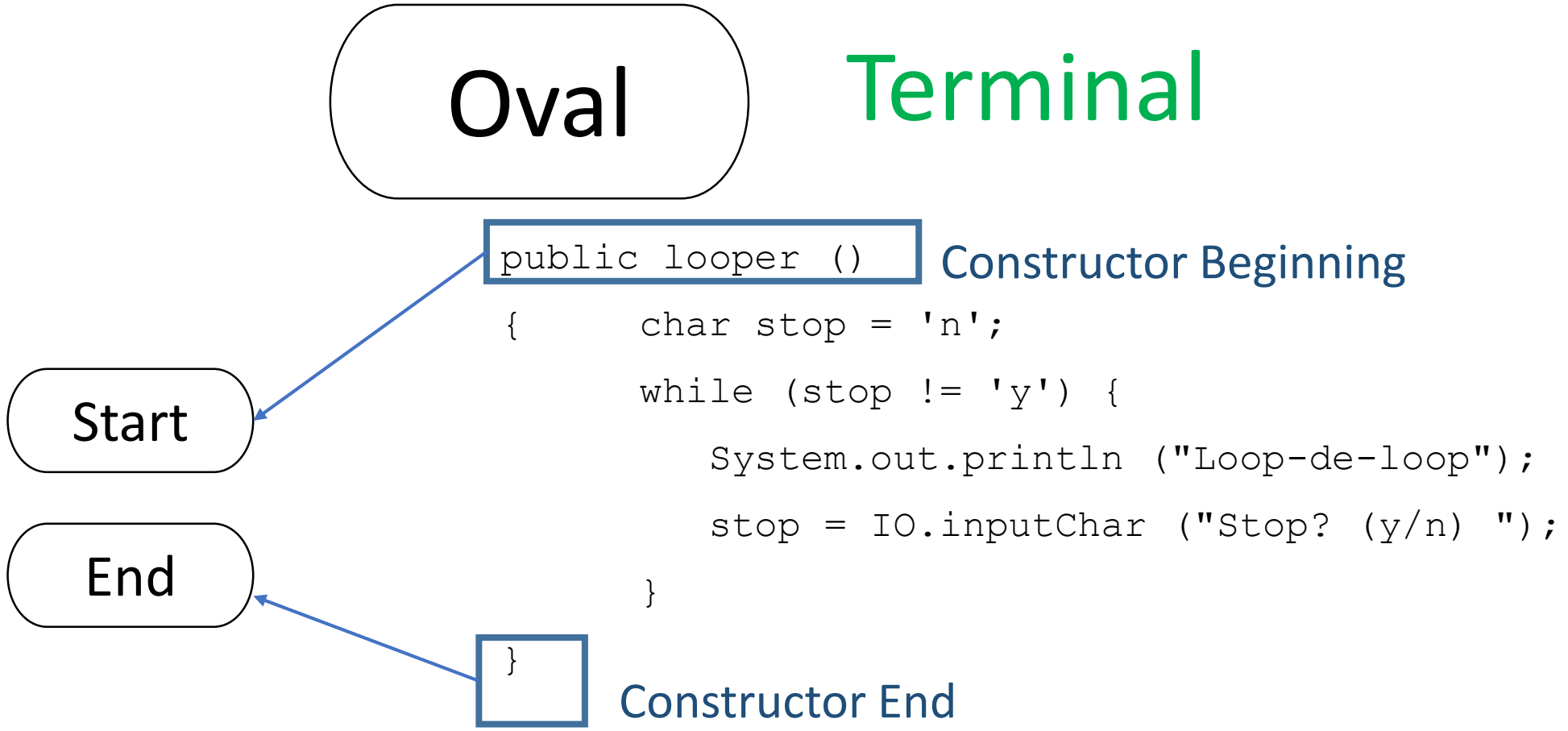
While Loop Flow Charts

Same great shapes, New great loop.

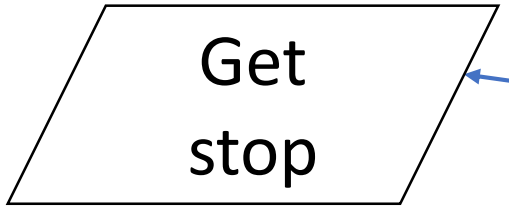


- One Start and One End for each program. No more, no less.
- Only holds “Start” and “End”
- Only one arrow comes out of the Start. No arrows for End.

Input

```
public looper ()
{       char stop = 'n';
        while (stop != 'y') {
            System.out.println ("Loop-de-loop");
            stop = IO.inputChar ("Stop? (y/n) ");
        }
}
```

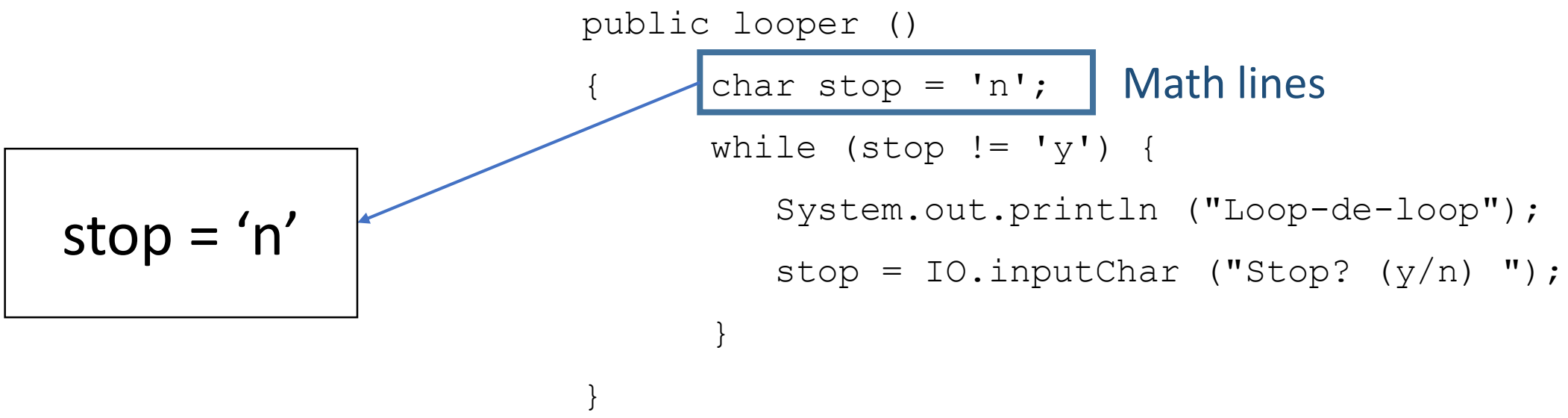
IO lines



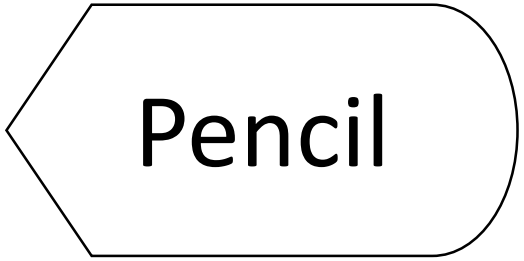
- Used for IO lines.
- Only write “Get” + variable name
- Don’t write the prompt (question)
- Only one arrow comes out of it.

Rectangle

Processing



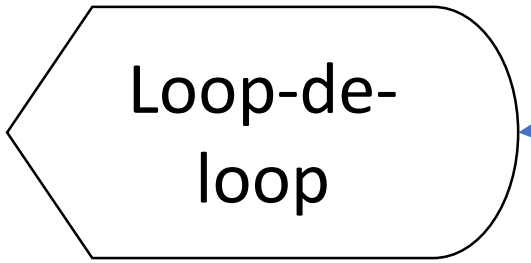
- Loop examples: int i=0; i++; i--; char stop=‘n’;
- Used for Math Lines.
- Leave out the variable type, but write everything else
- Only one arrow comes out of it.



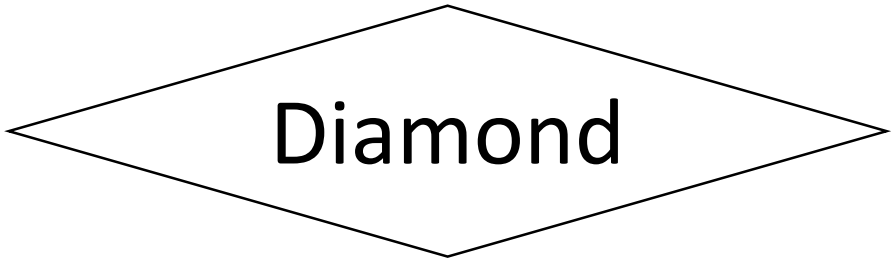
Output

```
public looper ()
{       char stop = 'n';
        while (stop != 'y') {
            System.out.println ("Loop-de-loop");
            stop = IO.inputChar ("Stop? (y/n) ");
        }
}
```

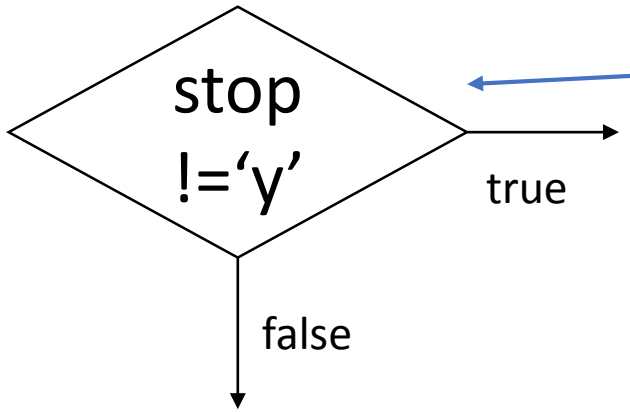
System.out lines



- Used for System.out.println lines.
- Leave out the System.out.println(); but write everything else
- Only one arrow comes out of it.



Boolean Expression




```
public looper ()
{       char stop = 'n';
        while (stop != 'y') {
            System.out.println ("Loop-de-loop");
            stop = IO.inputChar ("Stop? (y/n) ");
        }
}
```

- Examples: i<10; stop!=‘y; continu==‘y’; guess!=answer
- Used for Boolean Exressions – ifs && loops.
- Exactly 2 arrows come out of them. One is labelled true and one is labelled false.

Let’s start with a basic sandwich loop!

How does it start?

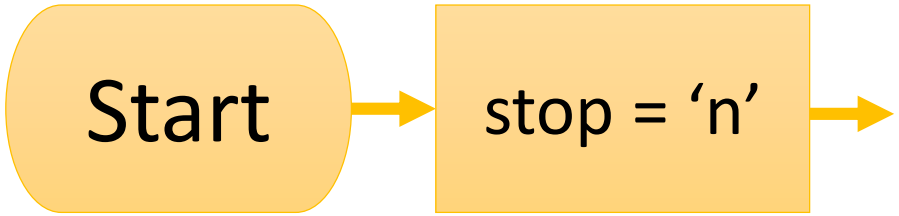
```
char stop = 'n';
while (stop == 'n')
{
    System.out.println ("Flowchart!");
    stop = IO.inputChar ("Stop? (y/n) ");
}
```

Then, we need to initialize the loop stopping variable.

```
char stop = 'n';
while (stop == 'n')
{
    System.out.println ("Flowchart!");
    stop = IO.inputChar ("Stop? (y/n) ");
}
```

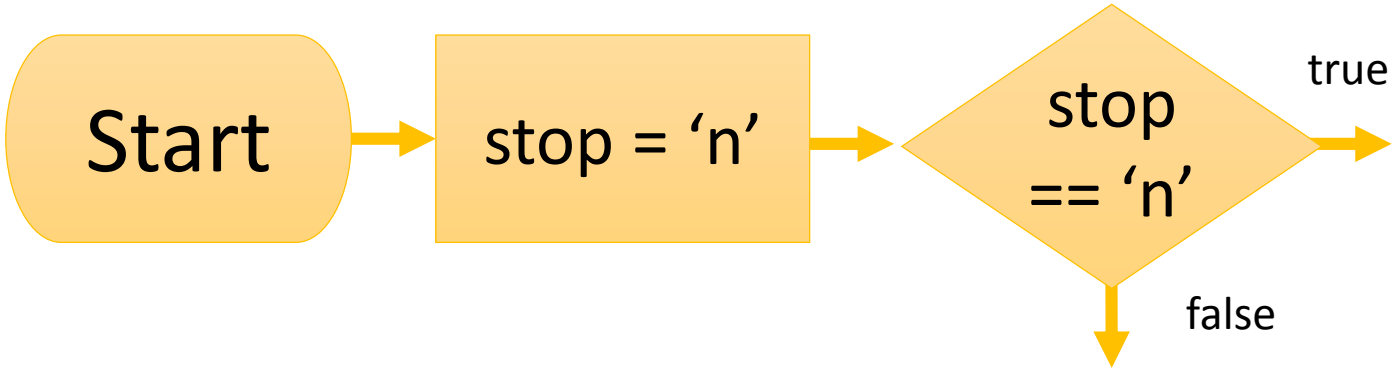
We need to handle the loop stopping condition.

```
char stop = 'n';
while (stop == 'n')
{
    System.out.println ("Flowchart!");
    stop = IO.inputChar ("Stop? (y/n) ");
}
```



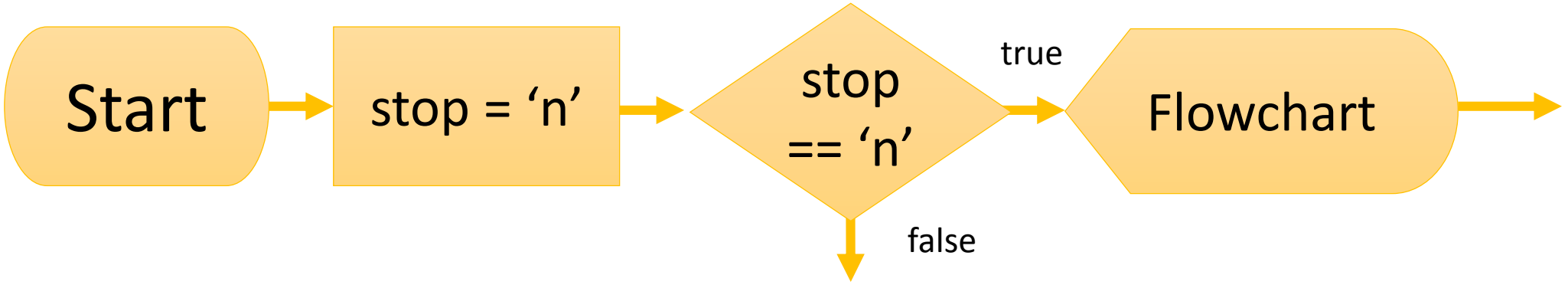
The next part is a step to repeat: System.out.println

```
char stop = 'n';
while (stop == 'n')
{
    System.out.println ("Flowchart!");
    stop = IO.inputChar ("Stop? (y/n) ");
}
```



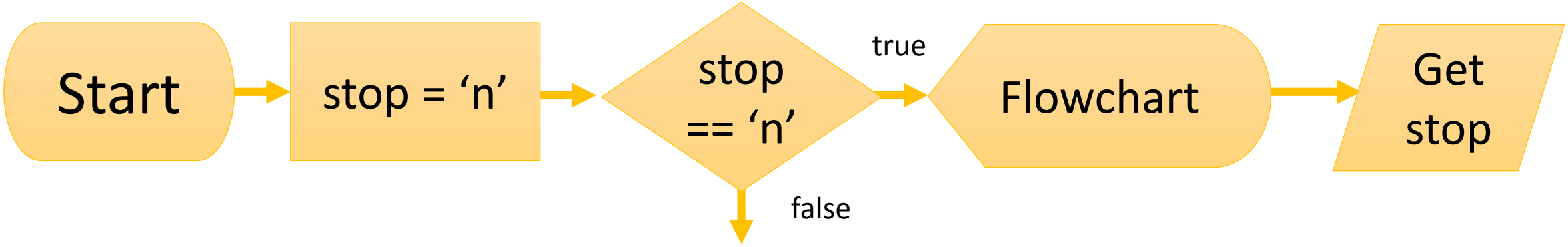
Then, we progress to the loop stopping condition

```
char stop = 'n';
while (stop == 'n')
{
    System.out.println ("Flowchart!");
    stop = IO.inputChar ("Stop? (y/n) ");
}
```



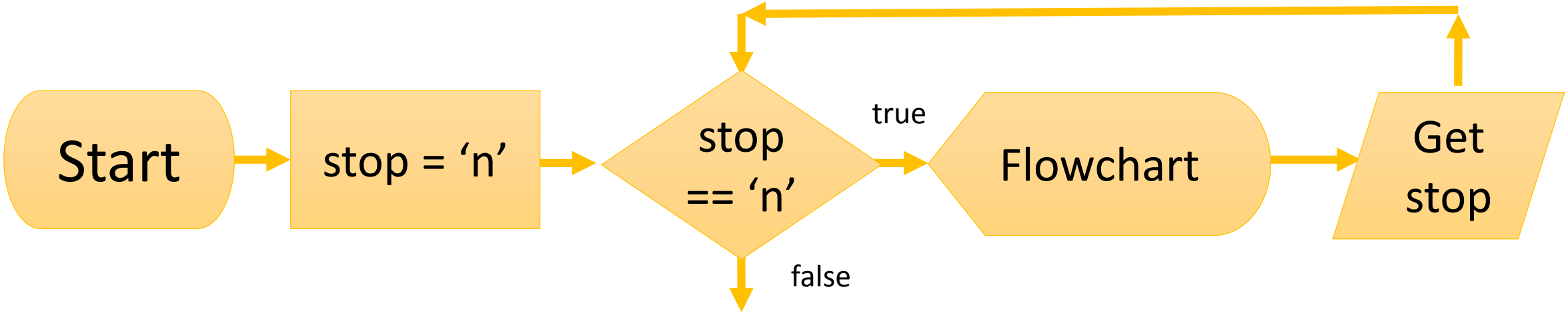
And loop back to the Boolean expression (aka Loop Stopping Condition)

```
char stop = 'n';
while (stop == 'n')
{
    System.out.println ("Flowchart!");
    stop = IO.inputChar ("Stop? (y/n) ");
}
```

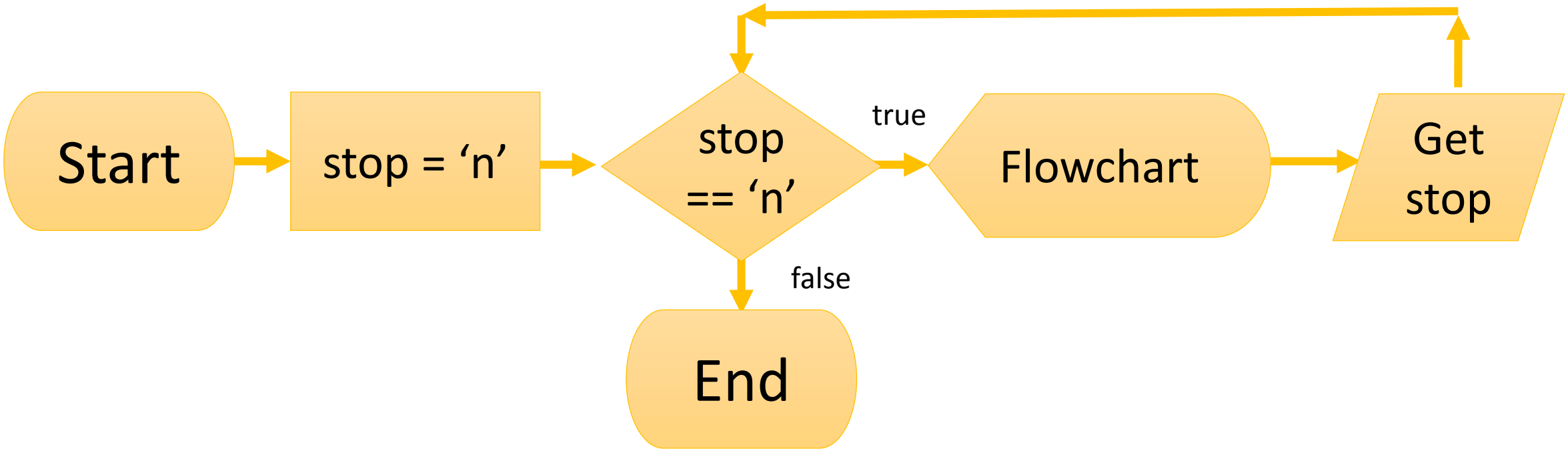


And how do we end?

```
char stop = 'n';
while (stop == 'n')
{
    System.out.println ("Flowchart!");
    stop = IO.inputChar ("Stop? (y/n) ");
}
```




```
char stop = 'n';
while (stop == 'n')
{
    System.out.println ("Flowchart!");
    stop = IO.inputChar ("Stop? (y/n) ");
}
```



Let's make a flow chart for this while loop!

How do we start?

```
int a = (int) (Math.random () * 10) + 1;
int b = (int) (Math.random () * 10) + 1;
int ans = IO.inputInt (a + "+" + b + "? ");
while (a + b != ans)
{
   System.out.println ("Nope. Again.");
   ans = IO.inputInt (a + "+" + b + "? ");
}
System.out.println ("You got it!");
```

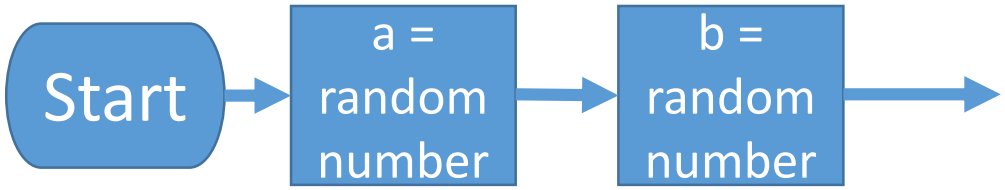
Then, we have two lines where we make new random numbers

Start

```
int a = (int) (Math.random () * 10) + 1;
int b = (int) (Math.random () * 10) + 1;
int ans = IO.inputInt (a + "+" + b + "? ");
while (a + b != ans)
{
   System.out.println ("Nope. Again.");
   ans = IO.inputInt (a + "+" + b + "? ");
}
System.out.println ("You got it!");
```

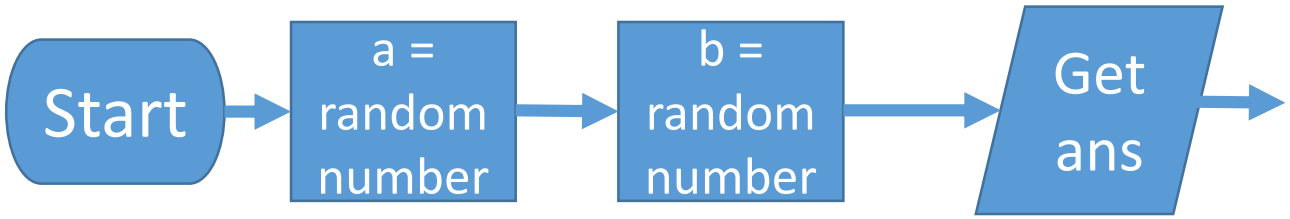
Then, we ask the user to add the numbers together

```
int a = (int) (Math.random () * 10) + 1;
int b = (int) (Math.random () * 10) + 1;
int ans = IO.inputInt (a + "+" + b + "? ");
while (a + b != ans)
{
   System.out.println ("Nope. Again.");
   ans = IO.inputInt (a + "+" + b + "? ");
}
System.out.println ("You got it!");
```



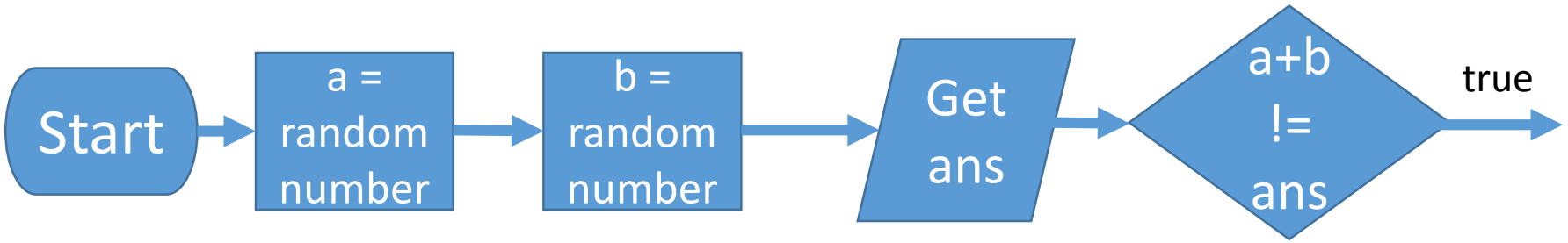
Then, we test and see if they were right!

```
int a = (int) (Math.random () * 10) + 1;
int b = (int) (Math.random () * 10) + 1;
int ans = IO.inputInt (a + "+" + b + "? ");
while (a + b != ans)
{
   System.out.println ("Nope. Again.");
   ans = IO.inputInt (a + "+" + b + "? ");
}
System.out.println ("You got it!");
```



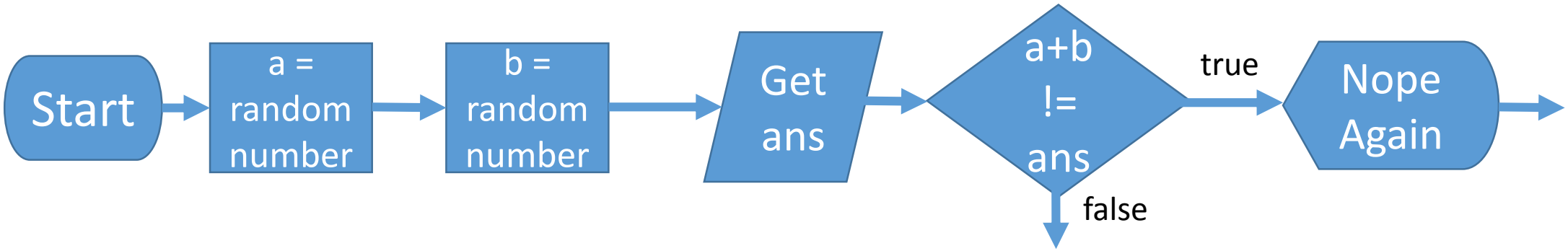
If they weren't right, we ask them to try again.

```
int a = (int) (Math.random () * 10) + 1;
int b = (int) (Math.random () * 10) + 1;
int ans = IO.inputInt (a + "+" + b + "? ");
while (a + b != ans)
{
   System.out.println ("Nope. Again.");
   ans = IO.inputInt (a + "+" + b + "? ");
}
System.out.println ("You got it!");
```



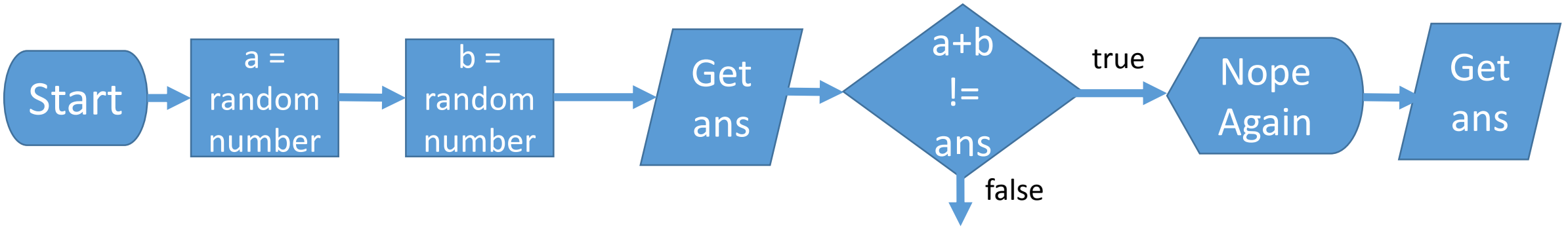
We get their new answer

```
int a = (int) (Math.random () * 10) + 1;
int b = (int) (Math.random () * 10) + 1;
int ans = IO.inputInt (a + "+" + b + "? ");
while (a + b != ans)
{
   System.out.println ("Nope. Again.");
   ans = IO.inputInt (a + "+" + b + "? ");
}
System.out.println ("You got it!");
```



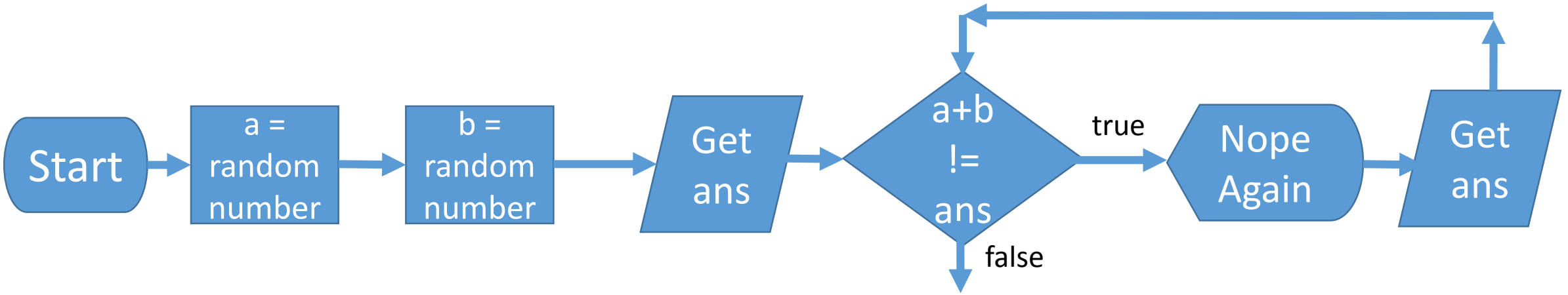
And we loop back

```
int a = (int) (Math.random () * 10) + 1;
int b = (int) (Math.random () * 10) + 1;
int ans = IO.inputInt (a + "+" + b + "? ");
while (a + b != ans)
{
   System.out.println ("Nope. Again.");
   ans = IO.inputInt (a + "+" + b + "? ");
}
System.out.println ("You got it!");
```



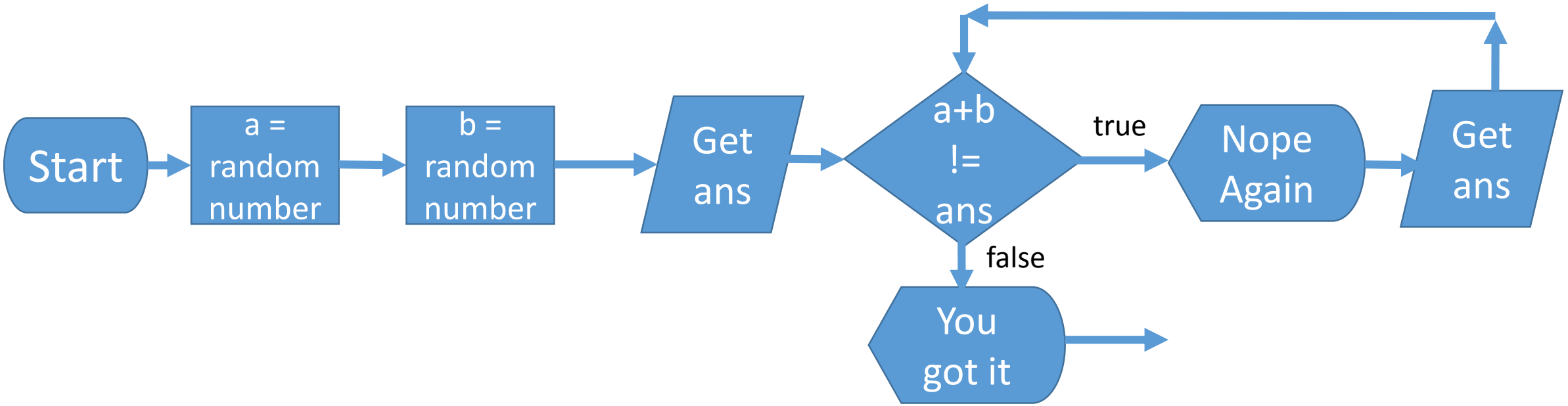
We tell them they got it right

```
int a = (int) (Math.random () * 10) + 1;
int b = (int) (Math.random () * 10) + 1;
int ans = IO.inputInt (a + "+" + b + "? ");
while (a + b != ans)
{
   System.out.println ("Nope. Again.");
   ans = IO.inputInt (a + "+" + b + "? ");
}
System.out.println ("You got it!");
```



And then we end

```
int a = (int) (Math.random () * 10) + 1;
int b = (int) (Math.random () * 10) + 1;
int ans = IO.inputInt (a + "+" + b + "? ");
while (a + b != ans)
{
   System.out.println ("Nope. Again.");
   ans = IO.inputInt (a + "+" + b + "? ");
}
System.out.println ("You got it!");
```




```
int a = (int) (Math.random () * 10) + 1;
int b = (int) (Math.random () * 10) + 1;
int ans = IO.inputInt (a + "+" + b + "? ");
while (a + b != ans)
{
   System.out.println ("Nope. Again.");
   ans = IO.inputInt (a + "+" + b + "? ");
}
System.out.println ("You got it!");
```