IT-University of Copenhagen

Automated software analysis

Analysis-1

Tróndur Høgnason (thgn@itu.dk)

October 11, 2017

1:

Prove that the following problem is undecidable - *Is boolean variable 'b' always true in program P?*
Assuming we have a program `bIsTrueInP(P)` that takes another program P as an input and decides whether or not the variable `b` is always true, we can construct the program R in Listing 1.

Listing 1: R.cs

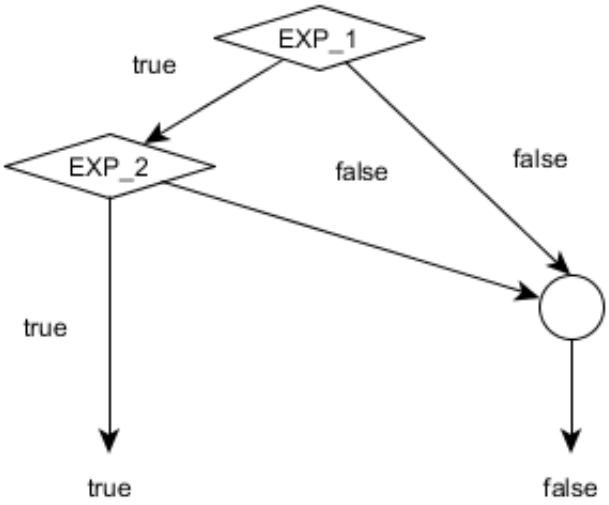
```
bool b = false;
P // insert the code of P here.
return b;
```

If we run `bIsTrueInP` with R as an input

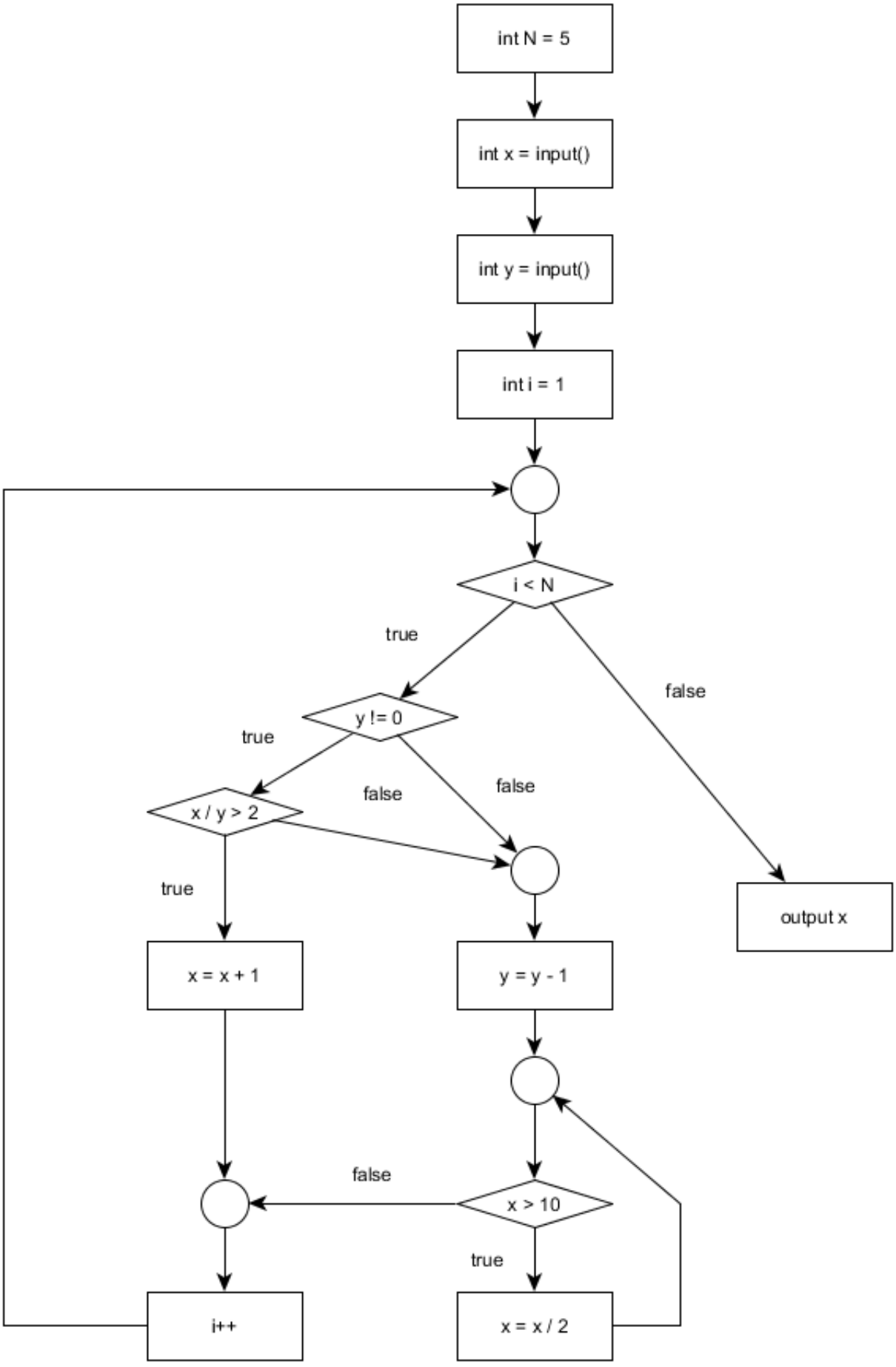
```
res = bIsTrueInP(R)
```

we can conclude that if `res` is true then the program R halts. If not, program R loops. This way we have solved the halting problem, which has been proven to be undecidable. We have therefore arrived at an contradiction. We must therefore conclude that the problem - *Is boolean variable 'b' always true in program P?* - is undecidable

2:



3:


int N = 5
int x = input()
int y = input()
int i = 1
i < N
true
false
y != 0
true
false
false
x / y > 2
true
output x
x = x + 1
y = y - 1
false
x > 10
true
i++
x = x / 2