Consider the program below. This program contains two nested loops. If you have access to the program's AST, than it's easy to see that: just look for two `while`-like statements. But, if you only have the CFG, how can you know that the program contains two loops?

```
01 int main(int argc, char** argv) {
02   int sum = 0;
03   int i = 1;
04   while (i < argc) {
05     char* c = argv[i];
06     while (*c != '\0') {
07       c++;
08       sum++;
09     }
10   }
11   printf("sum = %d\n", sum);
12 }
```

Figure 1: Program that computes the number of characters in the input line.

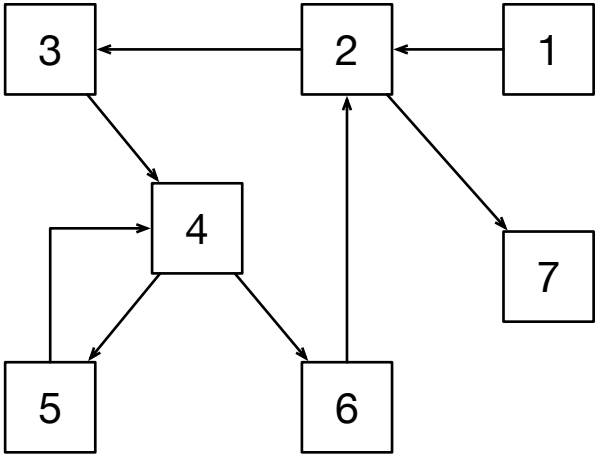

Figure 2: Control-flow graph of the program seen in Figure 1.

UFMG
DCC
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO
Compilers Laboratory

1. Can you relate the basic blocks in Figure 2 with program lines in Figure 1?

2. How do you know, looking only at the CFG in Figure 2, that the program would have two loops? Don't answer too fast! All that you can see in the CFG is a cycle involving five nodes!

3. How do you know which blocks refer to the innermost loop, and which blocks refer to the outermost loop?

4. Now, think about whatever you did to answer Q2 and Q3. Would your answer work if the CFG were undirected?