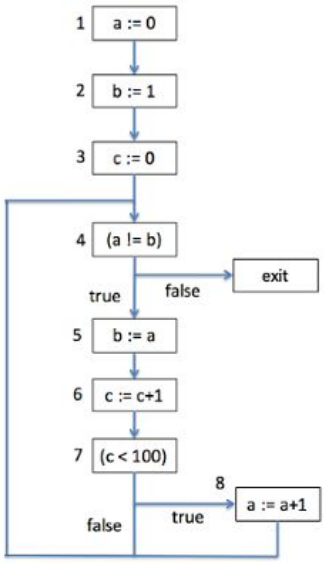
Dataflow Analysis

Question 1. Show the final result of performing reaching-definitions analysis on the given control-flow graph of the following program. That is, list the contents of IN[k] and OUT[k] for each k from 1 to 8.

```
a := 0; b := 1; c := 0;
while (a != b) {
    b := a;
    c := c + 1;
    if (c < 100)
        a := a + 1;
}
```



Answer:

Node	IN	OUT
1	⟨a, ?⟩, ⟨b, ?⟩, ⟨c, ?⟩	⟨a,1⟩, ⟨b, ?⟩, ⟨c, ?⟩
2	⟨a, 1⟩, ⟨b, ?⟩, ⟨c, ?⟩	⟨a, 1⟩, ⟨b, 2⟩, ⟨c, ?⟩
3	⟨a, 1⟩, ⟨b, 2⟩, ⟨c, ?⟩	⟨a, 1⟩, ⟨b, 2⟩, ⟨c, 3⟩
4	⟨a, 1⟩, ⟨b, 2⟩, ⟨c, 3⟩, ⟨a, 8⟩, ⟨b, 5⟩, ⟨c, 6⟩	⟨a, 1⟩, ⟨b, 2⟩, ⟨c, 3⟩, ⟨a, 8⟩, ⟨b, 5⟩, ⟨c, 6⟩
5	⟨a, 1⟩, ⟨b, 2⟩, ⟨c, 3⟩, ⟨a, 8⟩, ⟨b, 5⟩, ⟨c, 6⟩	⟨a, 1⟩, ⟨c, 3⟩, ⟨a, 8⟩, ⟨b, 5⟩, ⟨c, 6⟩
6	⟨a, 1⟩, ⟨c, 3⟩, ⟨a, 8⟩, ⟨b, 5⟩, ⟨c, 6⟩	⟨a, 1⟩, ⟨a, 8⟩, ⟨b, 5⟩, ⟨c, 6⟩
7	⟨a, 1⟩, ⟨a, 8⟩, ⟨b, 5⟩, ⟨c, 6⟩	⟨a, 1⟩, ⟨a, 8⟩, ⟨b, 5⟩, ⟨c, 6⟩
8	⟨a, 1⟩, ⟨a, 8⟩, ⟨b, 5⟩, ⟨c, 6⟩	⟨a, 8⟩, ⟨b, 5⟩, ⟨c, 6⟩

Question 2. Answer the following questions about dataflow analysis for the simple WHILE language from the lecture on dataflow analysis.

a. The order in which the nodes in the control-flow graph are processed by the inner loop "for (each node n)" of the chaotic iteration algorithm can affect which of the following aspects of the analysis?

A. Soundness B. Completeness C. Performance D. Termination

Answer: C

b. What is the most efficient order in which to visit the statements in the below program (i.e., nodes in its control-flow graph) for a reaching-definitions analysis?

S1; { if (...) then S2 else S3 }; S4

A. S1, S2, S3, S4 B. S4, S2, S3, S1 C. S1, S4, S2, S3 D. S2, S3, S1, S4

Answer: A

c. What is the most efficient order in which to visit the statements in the below program (i.e., nodes in its control-flow graph) for a live-variables analysis?

S1; { if (...) then S2 else S3 }; S4

A. S1, S2, S3, S4 B. S4, S2, S3, S1 C. S4, S2, S1, S3 D. S1, S2, S4, S3

Answer: B

d. Suppose the given program has N statements (i.e., nodes in the control-flow graph) and V variables. How much memory does a live-variables analysis consume on this program in the worst case at any instant? Write the size of memory as an expression in terms of N and V. Give as tight a bound as possible. Remember that the analysis must store both the "in" and "out" information for each statement.

Answer: 2 * N * V