Department of Electrical Engineering and Computer Science

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

6.035 Spring 2016

Test III

You have 50 minutes to finish this quiz.

Write your name and athena username on this cover sheet.

Some questions may be harder than others. Read them all through first and attack them in the order that allows you to make the most progress. If you find a question ambiguous, be sure to write down any assumptions you make. Be neat. If we can't understand your answer, we can't give you credit!

This exam is open book and open laptop. Additionally, you may access the course website, but aside from that you may NOT USE THE NETWORK.

Please do not write in the boxes below.

I (xx/24)	II (xx/40)	III (xx/36)	Total (xx/100)

Name:

Athena username:

I Lattices

Let P be the set of integer pairs whose each component can be either 1 or 2,

$$P = \{(1,1), (1,2), (2,1), (2,2)\}.$$

1. [8 points]: Define the relation $\preccurlyeq$ as:

$$(x_1, y_1) \preccurlyeq (x_2, y_2) \quad \text{if and only if} \quad x_1 \leq x_2 \textbf{ and } y_1 \leq y_2,$$
$$\text{here } \leq \text{ is the standard less-or-equal operator for integers.}$$

A. [4 points]: Is the relation $\preccurlyeq$ a partial order over P? If so, draw the Hasse Diagram. If not, write down a property that fails to hold.

B. [4 points]: Is the pair $(P, \preccurlyeq)$ a lattice? If not, write down a property that fails to hold.

2. [8 points]: Define the relation $\preccurlyeq$ as:

$$(x_1, y_1) \preccurlyeq (x_2, y_2) \quad \text{if and only if} \quad x_1 \leq x_2 \text{ or } y_1 \leq y_2.$$

A. [4 points]: Is the relation $\preccurlyeq$ a partial order over P? If so, draw the Hasse Diagram. If not, write down a property that fails to hold.

B. [4 points]: Is the pair $(P, \preccurlyeq)$ a lattice? If not, write down a property that fails to hold.

3. [8 points]: Define the relation $\preccurlyeq$ as:

$$(x_1, y_1) \preccurlyeq (x_2, y_2) \quad \text{if and only if} \quad x_1 = y_1 = 1.$$

A. [4 points]: Is the relation $\preccurlyeq$ a partial order over P? If so, draw the Hasse Diagram. If not, write down a property that fails to hold.

B. [4 points]: Is the pair $(P, \preccurlyeq)$ a lattice? If not, write down a property that fails to hold.

II Constant Analysis

In this question, we will perform an analysis on programs with a single integer variable, `x`, that determines if `x` is a constant value. Programs written in this language have five kinds of statements:

- `x = c;`

- `x = x + x;`

- `x = x * x;`

- `if (...) { ... } else { ... }`

- `while (...) { ... }`

In these statements, `c` is some integer constant.

To keep track of whether `x` is constant, we use the following flat lattice on integers.
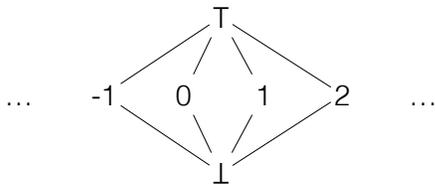
For example, if `x` is known to be a constant integer c, the analysis represents this information as c.

Before the analysis, we initialize the dataflow information at the entry point of the program to $\top$ and the information at all other program points to $\bot$.

At merge points, we update the dataflow information by computing the join of information from all incoming edges. The join operator $\vee$ computes the least upper bound of the lattice elements.

4. [4 points]: Compute the following:

$$\perp \vee \perp =$$

$$3 \vee 3 =$$

$$3 \vee 4 =$$

$$\top \vee 4 =$$

5. [4 points]: Is the lattice complete? Explain your answer.

6. [**4 points**]: What is the transfer function $f_n(e)$ for a statment n of the form x = c? Here e is the incoming dataflow lattice element. Your transfer function should correctly model the semantics of the program and be as precise as possible.

$f_n(e) =$

7. [**4 points**]: What is the transfer function $f_n(e)$ for a statment n of the form x = x + x? Here e is the incoming dataflow lattice element. Your transfer function should correctly model the semantics of the program and be as precise as possible.

$f_n(e) =$

8. [**4 points**]: What is the transfer function $f_n(e)$ for a statement n of the form x = x * x? Here e is the incoming dataflow lattice element. Your transfer function should correctly model the semantics of the program and be as precise as possible.

$f_n(e) =$

9. **[8 points]:**   Consider the following program.

```
x = 5;
if (...) {
  x = 1;
} else {
  x = 2;
}
x = x + x;
```

A. **[4 points]:** What is the analysis result at the end of this program?

B. **[4 points]:** What is the meet-over-paths solution for this program?

10. [8 points]: Consider the following program.

```
x = 5;
if (...) {
  x = 1;
} else {
  x = -1;
}
x = x * x;
```

A. [4 points]: What is the analysis result at the end of this program?

B. [4 points]: What is the meet-over-paths solution for this program?

11. [4 points]:   Does the analysis always terminate? If so, explain why. If not, give an example program or control flow graph where it fails to terminate.

III Data Dependence Analysis

12. **[18 points]:** Consider the following program:

```
FOR I = 1 to n
  FOR J = I to n
    A[I, J] = A[I+1, J+2] + 3
```

A. **[6 points]:** Assume that n = 4. In the grid below, circle the dots that represent the iteration space for this loop. Each dot represents the values of I and J for an iteration.

```
                    J

             1    2    3    4

        1    ●    ●    ●    ●

        2    ●    ●    ●    ●
   I
        3    ●    ●    ●    ●

        4    ●    ●    ●    ●
```

B. **[6 points]:** What is the distance vector for these loops?

C. **[6 points]:** Without any other optimizations or transformations, is either loop fully parallelizable as a FORALL loop? If so, is it the outer loop (I), the inner loop (J), or both, that can be parallelized?

13. **[18 points]:** Consider the following program:

```
FOR I = 1 to 5
  B[I] = B[I*2-1] + 3;
```

A. **[6 points]:** Assume you are using the integer linear programming method to find parallelizable loops. Write the set of linear inequalities for finding loop-carried data dependences of the above loop. You can use operators $\leq, \geq, <, >, =$, and $\neq$.

B. **[6 points]:** Write down all solutions of the linear programming inequalities.

C. **[6 points]:** Without any other optimizations or transformations, is the loop fully parallelizable as a FORALL loop? Explain your answer.