Roll No: _____

CS6843 Program Analysis at IIT Madras
MidSem Mar 5, 2014

Total Marks: 25 Number of questions: 9 **compulsory** questions
Duration: 90 minutes Marking: Q1 carries 1 mark, all others carry 3 marks each

1. Write the following sentence three times.
   "Rupesh, henceforth I will call you by your first name".

2. (a) Write the LLVM command for running your Shape Analysis for Assignment 3. [1 mark]

   (b) How many loads (memory accesses) does the following *getelementptr* instruction generate at runtime: *%idx2 = getelementptr i32* %MyVar, i64 1* ? _____ [1 mark]

   Why? [1 mark]

3. True or false? Do not give reasons.
   (a) Reaching definitions is a liveness property. _____

   (b) Live variables analysis is a liveness property. _____

   (c) Static analysis is sound for checking safety properties, but unsound for checking liveness properties.
   _____

4. Your analysis maintains only two bits XY (and nothing else) for the conditions $p \rightarrow null$ and $a \rightarrow x$ where $\rightarrow$ means "may point to". Initially, XY=10. You run your analysis on a program containing the following sequence of statements. Write the bit values of XY after each of these statements. Do not give reasons.

   (a) a = &x ____

   (b) x = &y ____

   (c) *p = *a* ____

5. For the following set of statements, compute Andersen's points-to information [0 marks]: **p = q; q = *a; x = *q; c = &x; q = &c; p = &a; x = &p;* And then answer the following.

   (a) Write *ptsto(x)*, that is, the final points-to set of *x*. _____

   (b) Write the set: *ptsto(p)* ∪ *ptsto(q)*. _____

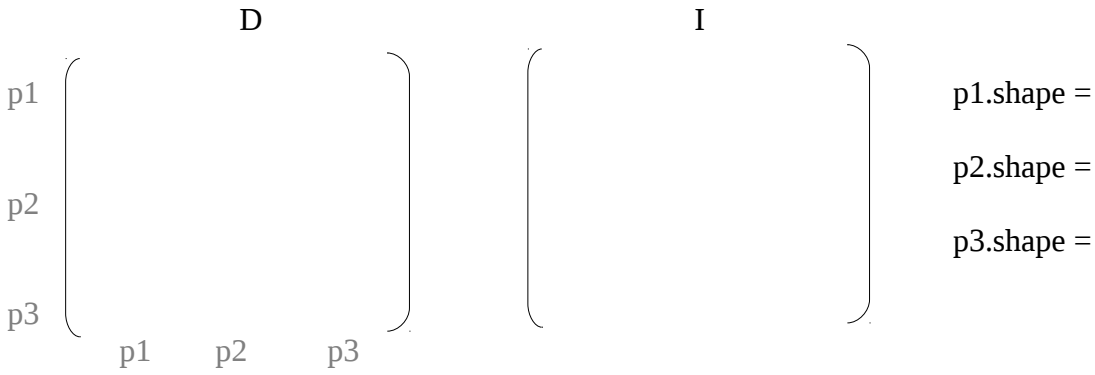
   (c) Write the set: *ptsto(a)* ∩ *ptsto(q)*. _____

6. For the set of statements in **Question 5**, consider a constraint graph-based formulation of points-to analysis.
   (a) Draw the constraint graph after you have evaluated all the constraints **once** and then propagated the information **once**.

   (b) Mention all the cycles in the final constraint graph.

7. For the set of statements in **Question 5**, draw the final Steensgaard's hierarchy.

8. If pointers *p1* and *p2* in a program are initially pointing to the same variable, while *p3* is pointing to some other variable, compute the D, I and shape values for *p1*, *p2*, *p3* after the following statement using Ghiya-Hendren's method: *p1→next = p3*.



9. **[Requires thinking. Try in the end.]** Devise a general strategy for improving precision of Andersen's Analysis A using Shape Analysis S. Thus, you have performed S, and are now performing A. How can you query S such that you can reduce the points-to information you compute using A? Illustrate with a concrete (and non-trivial) example.

Your marks in Program Analysis are neither going to shape up your future, nor they are going to matter. Whether your CGPA is 7.1 or 9.8 is going to be immaterial to your father-in-law. So don't worry, and Enjoy.