CS311 – Computational Structures – HW8

Assigned Thursday, November 29, 2012
will not be graded

Answer each question below. Write your answers neatly on paper. Be sure your name is on the paper, and the paper is clearly identified as Homework 9.

This assignment is based upon Chapters 11 and 12 of the Outline notes `http://web.cecs.pdx.edu/~sheard/course/CS311/Fall2012/notes/Outline.pdf`. This should give you some idea of what is important from the last few lectures, and what might appear on the final exam from these lectures.

1. Understand how to compute using the primitive recursive functions. Compute the following:
   (a) $P_2(1, 2, 3)$
   (b) $Z(21, 17, 42)$
   (c) $S(13)$
   (d) $(C\ S\ [Z])(1, 2, 3)$
   (e) $(\text{PR}\ Z\ Z)(3)$
   (f) $(\text{PR}\ Z\ (C\ S\ (P_1)))(3)$
2. Understand how the pairing and unpairing functions work. Compute the following
   (a) pair 5 6
   (b) pair 0 12
   (c) unpair 22
3. Understand that using the pairing functions data structures can be mapped to integers. Study the functions `eList` and `dList` as well as `ePR` and `dPR`.
4. Sketch the argument that there exists a total function that is not primitive recursive by using the diagonalization trick.
5. Develop trace trees for the following computations:
   (a) `(PR (C S [Z]) Z) [0]`
   (b) `(PR (C S [Z]) Z) [1]`

(c) `(PR Z (C S [Z])) [0]`

(d) `(PR Z (C S [Z])) [1]`

(e) `(PR (P 2) (P 3)) [1,2,3]`

Display your trace trees using the indentation convention used in the Outline.

6. Relate state traces of DFAs, configurations of PDAs, Instantaneous Descriptions of Turing Machines, and Trace trees of Primitive recursive functions.

7. Reproduce the proof that the Halting function is not computable. What kind of proof would you use?