

CSI 30 LECTURE NOTES (Ojakian)

Topic 3: Algorithms

OUTLINE

(References: 3.1)

1. Algorithms/programs

1. Algorithms

- (a) An algorithm is a finite set of precise instructions for performing a computation.
- (b) Example: multiplication, division, etc.
- (c) Example (dumb): Given any input positive integer, repeatedly subtract 1 till the value is zero, then stop and return the current value (i.e. zero ...)

2. Basic Programming Constructs

- (a) Use pseudo code or actual code.
- (b) Assignment statements
- (c) Basic data structure: Lists
- (d) Conditional statements
- (e) For Loops
- (f) While Loops
- (g) Loop Applications: Accumulators and Boolean Flags
- (h) Functions with return statement

3. Algorithms

PROBLEM 1. *Supposing x and y are set to integers. Set z to the larger of x or y .*

PROBLEM 2. *Write the definition of a function which takes an integer n as input and returns n if n is negative, 0 if n is zero, and $-n$ if n is positive (do with with a conditional statement, and without). Graph it.*

PROBLEM 3. *Write the definition of a function which takes two arguments, the first a negative integer s and the second a positive integer t . The function returns the sum of the even integers inbetween s and t (including s and t).*

PROBLEM 4. *Do one of the above for-loop questions with a while loop.*

4. Programs Instead of Proofs?

(a)

PROBLEM 5. *What do you think the sum of consecutive integers should be?
Write a program to test your hypothesis.*

PROBLEM 6. *Write program to test the claim: The sum of evens is even (test
all pairs of evens up to 100)*

(b) Do our programs prove the claim? What are the issues?

5. Harder Questions...

(a) Are there infinitely many prime numbers?

(b) Are there infinitely many twin primes?

(c) Test your hypothesis with a program.

6. Exercises

Section 3.1: 3, 4*, 13, 14, 16*, 19, 23, 24*

(where * means maybe hard ..)