Midterm Exam

Work alone. Do not use any notes or books. You have approximately 75 minutes to complete this exam.

Please write your answers on the exam. More paper is available if you need it. Please put your name at the top of each page.

1. Find the running time (20 points)

Below is an algorithm for testing if an array a with n elements contains three elements x, y, and z with x + y = z. Classify its running time in big- Θ terms.

```
for i <- 1 to n do
    for j <- 1 to n do
        for k <- 1 to n do
            if a[i]+a[j]=a[k] then
                return YES
return NO</pre>
```

Solution:

 $\Theta(n^3)$.

2. A simple program (20 points)

Below is a program written in the Random Access Language used in HW3. To refresh your memory, a succinct description of this language is given in the Appendix at the end of this exam. Assume that the memory initially contains the value 1 in all locations. Show the contents of locations 0 through 10 when the program halts.

0: LDA 0 1: ADD 0 2: STA 0 3: ADD 0 4: STA 0 5: STI 0 6: SUB 10 7: JMZ 9 8: JMP 4 9: HLT

Solution:

10 1

3. Binary numbers (20 points)

For each of the following decimal numbers, write it as a four-bit binary number in two's complement notation, or explain why doing so is not possible.

- 0=0000
- 1=0001
- -2=1110
- 7=0111
- 11=Too big to represent in four bits.

4. A simple digital circuit (20 points)

Draw a circuit using only 2-input AND gates that produces a 1 as output only if all four of its inputs are 1.

Solution:



5. Finding the median (20 points)

Describe an efficient algorithm for finding the k-th smallest number in an unsorted array of n distinct numbers, and classify its worst-case running time using big- Θ notation. Hint: you may use any algorithm you have seen in class as a subroutine.

Solution:

Version 1: using a sorting algorithm

- 1. Sort the numbers using your favorite sorting algorithm. Cost: $\Theta(n \log n)$ or $\Theta(n^2)$ depending on which algorithm you pick.
- 2. Return a[k].

Total cost: Same as for the sorting algorithm used in step 1.

Version 2: adaptation of selection sort

```
for i <- 1 to k do
    best <- i
    for j <- i to n do
        if a[j] < a[best] then
            best <- j
        swap a[best] with a[i]
return a[k]</pre>
```

This algorithm runs selection sort but stops when it has found the k smallest numbers. Total cost is $\Theta(kn)$, but in the worst case k = n so the cost is $\Theta(n^2)$.

Appendix

Random Access Language instructions

- ADD x Add the contents of location x to accumulator.
- SUB x Subtract the contents of location x from accumulator.
- LDA x Copy the contents of location x to accumulator.
- STA x Copy accumulator to location x.
- LDI x Copy location whose address is stored in x to accumulator.
- STI x Copy accumulator to location whose address is stored in x.
- JMP ℓ Jump to line ℓ of program.
- JMZ ℓ Jump to line ℓ of program if accumulator is zero.

HLT Halt.

CS 110 home page: http://zoo.cs.yale.edu/classes/cs110/ Fri 20 Dec 2002 16:32:23 EST exam
1.solutions.tyx Copyright © 2002 by Jim Aspnes