

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
```

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
```

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=
- 1=
- -2=
- 7=
- 11=

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.

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.

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.