BRONX COMMUNITY COLLEGE
of The City University of New York

DEPARTMENT OF MATHEMATICS and COMPUTER SCIENCE

CSI 30

Chapters 4 and 5 Review

1. Give an algorithm, using pseudocode, that takes a list of n distinct integers as input and finds the location of the largest odd integer in the list. If there is no such integer, the algorithm should return 0.

2. Present the algorithm, using a pseudocode, of finding the largest integer in an unordered sequence of n integers.

   Don't forget to describe the input and output for your algorithm.

3. Take a look at the following algorithm:

```
procedure it(n:positive integer)
sum := 0
For i:= 1 to n
    sum := sum + i*i
End-for
Return(sum)
```

   (a) How many multiplication operations will be done (an expression with `n`)?

   (b) If $n = 3$, what value will be returned?

4. Given the algorithm:

```
procedure thing(a_1,a_2,a_3,...a_n:integers)
sum1 := 0
sum2 := 0
For i := 1 to n
   If (a_i > 0), sum1 := sum1 + a_i
   If (a_i < 0), sum2 := sum2 + a_i
End-for
Return(sum1,sum2)
```

   For the set of values {1,5,-2,-9,2,5,-7} as input for the above algorithm, what are values of `sum1` and `sum2` that will be returned?

5. Consider the algorithm:

```
procedure foo(n:integer)
If n > 10, print('A')
If (n <= 10) and (n > -10), print('B')
Else print('C')
```

a) What will be printed if the procedure `foo` is run on `n=4`?

b) What will be printed if the procedure `foo` is run on `n=-4`?

c) What will be printed if the procedure `foo` is run on `n=24`?

6. Use Linear search to find 13 in the following list: $1, 7, 2, 3, 6, 8, 13, 4, 89$
   How many comparisons will be performed?

7. Can binary search be used if it gets the following list as input: $1, 7, 2, 3, 6, 8, 13, 4, 89$?

8. Use binary search to find 14 in the following list: $1, 6, 8, 9, 13, 14, 16, 22, 36, 38$
   Show all the splits and all the middle elements.

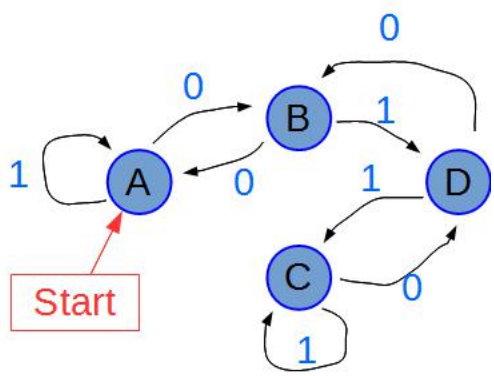
9. If binary search is used to find 10 in the following list: $9, 10, 14, 16, 22, 36, 56, 59, 61$
   How many splits will be performed before the element is found?

10. Use **bubble sort** algorithm to sort the list $5, 2, 4, 1, 3$
    (show all the passes, with interchanges, see our lecture slides)
    How many interchanges will be performed during the first pass?

11. Use **insertion algorithm** to sort the list $5, 2, 4, 1, 3$
    (show all shifts/insertions, see our lecture slides).
    When the algorithm reaches value 1,at position 4, i.e. `j=4`, what elements will be shifted one space to the right?

12. Use **selection algorithm** to sort the list $5, 2, 4, 1, 3$
    (show all swaps and passes, see our lecture slides).
    How many swaps are done for the given sequence during the entire sorting procedure?

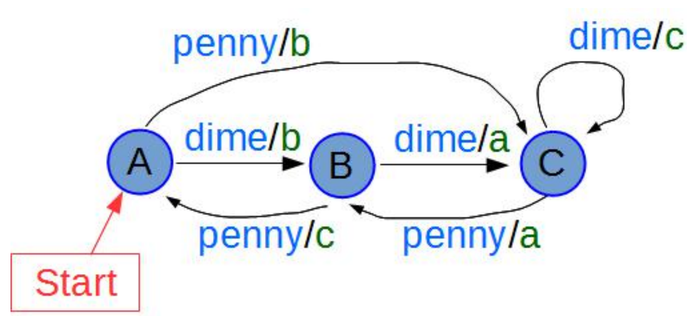
13. For the following FSM



1) What is the current state after the FSM has processed the input sequence `0 1 0 1 1`?

2) What input sequence required to get from state A to state A and changing to at least one other state?

14. For the following FSM



1) What is the current state after the FSM has processed the input sequence `PENNY DIME PENNY DIME`?

2) What input sequence required to get from state A to state A and changing to at least one other state?

15. The following Turing machine subtracts two unary numbers (i.e. numbers that consists only of 1s). For example, 7-5 = 1111111 - 11111 = 11 = 2

Input format: $*$ (blank) in the beginning to mark the first number, then

sequence of 1s representing the first number, then

$*$ (blank) separating the two numbers, then

sequence of 1s representing the second number, followed by blanks.

Example: $*11111*11***\ldots$

For the difference $a - b$:

if $a < b$, then Turing machine *rejects* the input.

if $a >= b$, then the 1s left on the tape represent the answer, and the Turing machine *accepts* the input.

If the first number the machine starts is not blank ($*$), the machine *rejects* the input as well.

$S = \{q_0, \ldots, q_5, q_{acc}, q_{rej}\}$, $\Gamma = \{0, 1, *\}$

	1	0	$*$	
q_0	$(q_{rej}, 1, R)$	$(q_{rej}, 0, R)$	$(q_1, *, R)$	simply steps over the first blank
q_1	$(q_1, 1, R)$	$(q_1, 0, R)$	$(q_2, *, R)$	runs till the end of first number
q_2	$(q_2, 1, R)$	$(q_3, 0, L)$	$(q_3, *, L)$	runs till the end of the second number
q_3	$(q_4, 0, L)$	$(q_3, 0, L)$	$(q_{acc}, *, L)$	decrements one from second number
q_4	$(q_4, 1, L)$	$(q_4, 0, L)$	$(q_5, *, L)$	runs to the first number
q_5	$(q_1, 0, R)$	$(q_5, 0, L)$	$(q_{rej}, *, R)$	decrements one from the first number

comments: If in state q_3, 1 is met - we change it to 0 (this 1 is being subtracted), and move on to state q_4

If in state q_3, $*$ is met, it means that all 1s from the second number are already subtracted, we finished.

If in state q_3, 0 is met, it means that this 1 is already being subtracted, move on to the next one to subtract.

If in state q_5, blank ($*$) is met, it means that the number that is being subtracted is larger, therefore we need to reject the input.

Show what will be left on the tape after Turing machine processes it and what will the final state of the machine.

$*$	1	1	1	1	$*$	1	1	$*$	$*$	...

Answer:
