COS 126 Written Exam 2 Solution, Spring 2008

1. **Binary Search**. The statements below explain why using binary search makes sense.

   A. It is typically much faster than sequential search.

   C. When data is sorted, it is easy to cut the amount of data to search in half.

2. **Postfix**. `6 5 4 * 3 2 1 + * + +`

```
6 20    3 2 1 + * + +
6 20    3 3     * + +
6 20    9         + +
6 29                +
35
```

3. **Regular expressions**. The strings below are in the language described by the regular expression:

   `a*bb(ab|ba)*`

   `A. abb`

   `D. bbbaab`

4. **Linked structures**.

   a) does not match any of the methods. Note that a call with the number "7" would infinite loop for method f().

   b) f()

   c) g()

   d) h()

5. **OOP terminology**.

a) 3 instance variable declaration

b) 4 constructor signature

c) 19, 20 object creation

d) 6, 11 overloaded method signature

e) 21, 22 method invocation or call

f) 3, 18 primitive type variable declaration

g) 19, 20 reference type variable declaration

6. **Circuits**.

a) The truth table.

a	b	c	x
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

b) Boolean Expressions equivalent to the circuit.

`b. ab' + bc'`

`e. a'bc' + ab'c' + ab'c + abc'`

7. **Turing machines.**

   A. X - doesn't move along the tape

   B. X - works only for 1*0*

   C. correct

   D. correct

8. **Theory**.

   A. T/F Any RE with closure (a "*" or a "+") describes infinitely many strings.

   B. T Any RE without closure describes only finitely many strings.

   C. T Given unlimited memory, TOY can solve any problem that can be solved by a TM.

   D. T It is possible to write a program that detects some Java programs that terminate.

   E. F It is possible to write a program that detects all Java programs that terminate.

   F. T If P=NP, there is no problem that is NP but is not NP-complete.

   G. T A Turing machine can decide whether a given string is in the language described by a given regular expression.

   H. F No Turing machine can decide whether a given DFA halts.

   I. F The undecidability of the halting problem is a statement about Turing machines: it is *not* applicable to real computers.

   J. T The Church-Turing thesis is a theory about our universe: it cannot be proven mathematically.

   K. T If P equals NP, then the Traveling Salesperson Problem can be solved in polynomial time by a deterministic Turing Machine.

   L. F If P does not equal NP, then there is no case of the Traveling Salesperson Problem for which you can find the optimal tour in polynomial time.

   M. F Factoring is known to be in NP but has not been proven to be NP-complete, so the discovery of a polynomial-time algorithm for factoring would mean that P equals NP.

   N. F Factoring is known to be in NP but has not been proven to be NP-complete, so the discovery of a polynomial-time algorithm for TSP would still not give a polynomial time algorithm for factoring.