| COS 126 | General Computer Science | Spring 2009 |
|---|---|---|

# Exam 2 Solutions

1. **Data types.**

   (a) A set of values and operations defined on those values.
   *This mantra was repeated in several lectures.*

   (b)

   | | |
   |---|---|
   | **A** Hides the instance variable from code in other files. | A. `private` |
   | **A** Hides the method from code in other files. | B. `public` |
   | **A** Hides the subclass from code in other files. | C. `final` |
   | **B** Exposes the API method to code in other files. | D. `static` |
   | **C** Prevents the value of the instance variable from being changed once initialized. | E. none of the above |

   (c)

   - To limit scope (which makes programs easier to test and debug).
   - To help enforce encapsulation of the data type (hide the internal representation).
     - Enables substituting an improved implementation (simpler, faster, more accurate) of the data type without changing the client because the client code won't depend on internal representation, e.g., to avoid Y2K time bombs.
     - To ensure data type value remains in a consistent state because client code can only access the data type value through the public API.

2. **Scientific computation.**

   III.
   *As indicated in lecture, there are more complicates formulas that are accurate for all distances.*

3. **Regular expressions.**

```
String re = "(ATG)((A|C|T|G)(A|C|T|G)(A|C|T|G))*(TAG|TAA|TTG)";
```

4. **Pass-by-value and references.**

```
1
2222
6
9999
```

5. **Analysis of algorithms.**

   (a) 750 seconds.
   The running time is proportional to $N^3$—doubling the input size increases the running time by a factor of 8.

   (b)

   - If the constant in the quadratic algorithm is significantly lower than the constant in the linearithmic algorithm, the quadratic algorithm may run faster for the values of $N$ you are interested in solving.
   - The quadratic algorithm may be substantially easier to implement, test, and debug.
   - The quadratic algorithm may be easier to understand and prove correct.
   - The quadratic algorithm may use less memory.

6. **Stacks and queues.**

```
public String dequeue() {
    if (isEmpty()) throw new RuntimeException("Queue underflow");
    String item = first.item;
    first = first.next;
    return item;
}

public void enqueue(String item) {
    Node x = new Node();
    x.item = item;
    if (isEmpty()) { first = x;     last = x; }
    else           { last.next = x; last = x; }
}
```

7. **Theory of computation.**

**I** Princeton computer science professor proves P = NP; existence of polynomial-time algorithm for factoring (and breaking the RSA cryptosystem) remains an open question.

**I** Symantec Releases 100% guaranteed general-purpose virus detector.

**P** Princeton student solves SAT instances that have thousands of variables in their senior thesis.

**P** Princeton graduate student publishes polynomial-time algorithm for factoring integers—banking industry in an uproar.

**I** COS 126 student proves that $P = NP$ by experimentally verifying that the smallest-insertion heuristic runs in $N^2$ time.

**I** COS 126 preceptor develops software to decide program equivalence—claims it will completely automate grading of programs.

**I** Physicist identifies a search problem that can be solved in polynomial time on a classical computer, but not on a Turing machine.

**P** COS 126 student finds provably-optimal TSP tour for `tsp1000.txt`.

8. **Circuits.**

(a)

$(X + Y)'(XY)(Z + X) + Z$

(b)

| $X$ | $Y$ | $Z$ | $f$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 |

(c)

$f(X, Y, Z) = Z$