Assignment 2

Due date: 10:00am, Monday, 18th of August 2014

General Instructions

You have to do this assignment as a **team of two students** if you are an undergraduate student, or **individually** if you are a postgraduate student. Individual work of an undergraduate student is also allowed and will be treated same as team work. Team members should be from the same tutorial group. All implementations have to be done in JAVA.

Submissions have to include coversheet including names, student ids, and your tutorial group such that submissions can get marked.

Submit your solutions (including printout of the source code) for Exercises 1, 2, and 3 as well as the results of the execution of your programs to the box "ADSA" on level 4, Innova21 (close to reception) **by the deadline**. **No late submissions will be accepted.**

In addition, submit your source code for Exercises 2 and 3 using the web submission system.

Exercise 1 *Recurrence (4 + 2 points)*

1. Assume that a is contant and so is $T(n)$ for $n \leq a$. For recurrence $T(n) = T(n - a) + T(a) + n$, prove that $T(n) = O(n^2)$.
2. Is the above bound tight if $a = \frac{n}{2}$ (non-constant)? If yes, explain why. If not, give a better bound.

Exercise 2 *Computation of Inversions (CLRS Problem 2-4) (5+2 points)*

Let $A[1..n]$ be an array of n distinct numbres. If $i < j$ and $A[i] > A[j]$, then the pair (i, j) is called an *inversion* of A.

1. Design and implement an algorithm that determinies the number of inversions in any permutation on n elements from the set {1, 2, ..., n} in $\Theta(n \log n)$ worst-case time. (Hint: Modify merge sort.) Test your program for the permutations {10, 1, 2, 3, 4, 5, 6, 7, 8, 9}, {1, 5, 2, 6, 4, 3, 8, 9, 7, 10} and {10, 9, 8, 7, 6, 5, 4, 3, 2, 1}.
2. What permutation has the most inversions? How many does it have?

Exercise 3 *Polynomial Multiplication (2+5 points)*

We are given two polynomials $p(x) = \sum_{i=0}^{n-1} p_i x^i$ and $q(x) = \sum_{i=0}^{n-1} q_i x^i$, where $n = 2^k$. The following algorithm for polynomial multiplication is conceptually similar to Karatsuba's algorithm of integer multiplication:

- Let $p(x) = \sum_{i=0}^{\frac{n}{2}-1} p_i x^i + \sum_{i=\frac{n}{2}}^{n-1} p_i x^i = p_1(x) + p_2(x)x^{\frac{n}{2}}$ and $q(x) = \sum_{i=0}^{\frac{n}{2}-1} q_i x^i + \sum_{i=\frac{n}{2}}^{n-1} q_i x^i = q_1(x) + q_2(x)x^{\frac{n}{2}}$.
- Recursively compute $A(x) = p_1(x)q_1(x)$, $C(x) = p_2(x)q_2(x)$, $B(x) = (p_1(x) + p_2(x))(q_1(x) + q_2(x)) - A(x) - C(x)$.
- Return $r(x) = p(x)q(x) = A(x) + B(x)x^{\frac{n}{2}} + C(x)x^n$.

1. Analyse the time complexity of the above algorithm.
2. Implement the above algorithm. Your program should ask for the input n and the two arrays of coefficiencies p and q. The output should be an array of coefficiencies q of the resulting polynomial $q(x) = p(x)q(x)$.

   Test your program for the input arrays:

   (a) $p = [5, 0, -6, 8, 3, -9, 0, 8]$ and $q = [0, 10, -9, 8, -3, 0, 0, 9]$;

   (b) $p = [-9, -7, 0, 0, 7, 9, 8, -7, 0, -3, 0, 8, 0, -3, 0, 2]$ and $q = [0, 0, 8, -2, 0, 3, -5, 4, 0, -6, 1, -9, 6, 3, 8, 0]$.

   Record the execution of the program for the submission of the assignment.

Submission instructions for programming code

First, type the following command, all on one line (replacing `aXXXXXXX` with your username):

```
svn mkdir --parents -m "ADSA"
https://version-control.adelaide.edu.au/svn/aXXXXXXX/2014/s2/adsa/assignment2
```

Then, check out this directory and add your files:

```
svn co https://version-control.adelaide.edu.au/svn/aXXXXXXX/2014/s2/adsa/assignment2
cd assignment2
svn add File1.java
svn add File2.java
...
svn commit -m "assignment2 solution"
```

Next, go to the web submission system at:

https://cs.adelaide.edu.au/services/websubmission/

Navigate to *2014*, *Semester 2*, *Adelaide*, *Algorithm and Data Structure Analysis*, then *Assignment 2*. Click *Make A New Submission For This Assignment* and indicate that

you agree to the declaration. The automark script will then check whether your code compiles. You can make as many resubmissions as you like. If your final solution does not compile you won't get any marks for this solution.

The websubmission system for this assignment will open till 10am Monday August 18th, 2014.

End of Questions