CPSC 320 Another Sample Midterm 1
October 2014

Name: ____________________ Student ID: ____________________

Signature: ____________________

- You have 50 minutes to write the 5 questions on this examination.
  A total of 40 marks are available.
- **Justify all of your answers.**
- You are allowed to bring in one hand-written, double-sided 8.5 x 11in sheet of notes, and nothing else.
- Keep your answers short. If you run out of space for a question, you have written too much.
- The number in square brackets to the left of the question number indicates the number of marks allocated for that question. Use these to help you determine how much time you should spend on each question.
- Use the back of the pages for your rough work.
- **Good luck!**

Question	Marks
1	
2	
3	
4	
5	
Total	

UNIVERSITY REGULATIONS:

- Each candidate should be prepared to produce, upon request, his/her UBC card.
- No candidate shall be permitted to enter the examination room after the expiration of one half hour, or to leave during the first half hour of the examination.
- CAUTION: candidates guilty of any of the following, or similar, dishonest practices shall be immediately dismissed from the examination and shall be liable to disciplinary action.
  1. Having at the place of writing, or making use of, any books, papers or memoranda, electronic equipment, or other memory aid or communication devices, other than those authorised by the examiners.
  2. Speaking or communicating with other candidates.
  3. Purposely exposing written papers to the view of other candidates. The plea of accident or forgetfulness shall not be received.
- Candidates must not destroy or mutilate any examination material; must hand in all examination papers; and must not take any examination material from the examination room without permission of the invigilator.

[12] 1. Short Answers

[3] a. Why is it useful to know that the furthest-in-the-future cache maintenance algorithm is optimal, even though it can not be implemented in practice?

[3] b. Mr. Isulo, the well-known alien computer scientist, draws a decision tree for the `MarvinSort` algorithm for $n = 10$ and shows it to you. What information can you infer from this tree (even though you have never heard of `MarvinSort`)?

[3] c. Is the Gale-Shapley algorithm greedy? Explain why or why not.

[3] d. Is $n/100 \in o(n)$? Explain why or why not.

[5] 2. A computer scientist who believes both genders are created equal wants to modify the Gale-Shapley algorithm as follows:

- In each iteration, either a free man or a free woman proposes to the first person on his/her preference list that he/she has not yet proposed to.
- The algorithm terminates when no free man or woman remains.

Give an example that proves that the matching produced by this modified version of the algorithm is not necessarily stable.

[6] 3. Write a recurrence relation that describes the worst-case running time of the following algorithm as a function of n. You may ignore floors and ceilings. Note: I do not believe that this algorithm computes anything useful, so don't waste any time trying to understand what it does.

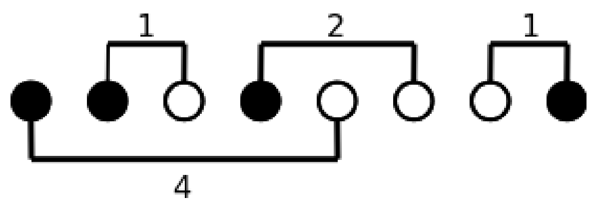
```
Algorithm BugsB(A, first, n)
    if n < 3 then
        return A[first] - 1
    endif

    x ← BinarySearch(A, first, n, BugsB(A, first + ⌊n/3⌋, ⌊n/3⌋))
    ElmerJ(A, first, n, x)
    return BugsB(A, first, ⌊n/6⌋) * BugsB(A, first + ⌊n/2⌋, ⌊n/4⌋)
```

Assume that the call `ElmerJ(A, first, n, x)` runs in $\Theta(n \log n)$ time.

[11] 4. Consider the following problem: there are $2n$ dots that are equally spaced on a line. Each dot is either white or black, and there are n dots of each color. The dots of a given color are not necessarily together; colors are interleaved arbitrarily. For instance:

We want to match each black dot with a white dot, while minimizing the sum of the distances between the elements of each pair. For instance, if we pair the dots of the previous figure as follows:



then the sum of the distances is $4 + 1 + 2 + 1 = 8$.

[8] a. Describe a greedy algorithm to find the matching that minimizes the sum of the distances between the elements of each pair. The input to your algorithm should be an array `C` of colors; for the example we would have `C[0] = black`, `C[1] = black`, `C[2] = white`, etc.

[3] b. Analyze the time complexity of your algorithm from part (a).

[6] 5. In class, we proved the following theorem about the furthest in the future algorithm:

> If S_j is a reduced schedule that is the same as S_{FF} for the first j requests, then there is a reduced schedule S_{j+1} that is the same as S_{FF} for the first $j+1$ requests, and incurs no more misses than S_j in total.

[3] a. Does the schedule S_j need to be optimal for this theorem to work? If so, why? If not, what happens if S_j is not optimal?

[3] b. In the third case of the proof, we obtained S_{j+1} by first changing the element that S_j was evicting in response to the $(j+1)^{\text{st}}$ request. Why could we not then write that S_{j+1} would make the same decisions as S_j from request $j+2$ onwards (and stop with that statement)?