CS 32

Exam 1 - Answer Key

February 18, 2015

General Instructions

- Answer the items completely. Show your solutions/justifications when asked.
- Write as legibly as possible. Illegible or unreadable answers and solutions may not merit any points.
- Refrain from making unnecessary motions and sounds during the exam. Any suspicious behavior will be dealt with accordingly.
- Direct all questions to the proctor.
- If you need to go to the CR, hand your questionnaire, answer sheet, and scratch paper to the proctor before heading out. Only one person at any given time is allowed to go out.
- Once you're done with the exam (one way or the other), place your scratch papers and the questionnaire inside your blue book.

Questions

1. Given a multistack $\mathbb{MS}$ with m stacks implemented on an array of size n, what is the time complexity of the part of the Garwick-Knuth algorithm wherein the new bottoms of the stacks have to be computed? Justify your answer. *(1 point)*

   ANSWER: Assuming, each statement has cost = 1, here is an analysis of the relevant parts of the Garwick-Knuth algorithm for the purpose of computing the new bottoms of the stacks:

   - Computing for the number of free cells entails getting the number of elements per stack has to be obtained (via subtracting the current top index and the current bottom index of each stack, done in constant time), which has to be done m times, and then subtracting it from n. Total running time is m + 1 steps.
   - Computing for total growth/increment is done by getting the difference between the old and new top indices per stack, which also has to be done m times.The total running time for this is m steps.
   - Computing for allocation factors (a and b) is takes one step each, for a total running time of 2 steps.
   - Computing for $s, t, NEWB$ (each of which takes constant time) has to be done m times. Therefore, the running time is 3m steps, assuming the final check (i.e. computing for $NEWB(m+1)$ is no longer made. *(NOTE: This is actually the most expensive part of the whole process, so it is sufficient that the analysis be focused on this part alone.)*

   The total running time therefore is $m + 1 + m + 2 + 3m = 5m + 3 \in O(m)$.

2. Four stacks coexist in an array of size 600. The state variables of the stacks are as follows:

   - B(1:5) = (0, 181, 310, 450, 600)
   - T(1:4) = (140, 310, 400, 510)
   - OLDT(1:4) = (170, 261, 360, 535)

   An item is pushed onto the 2nd stack, causing an overflow. An attempt was made to reallocate free cells using the Garwcik-Knuth algorithm, the trail of computation of which is as follows:

   - $freecells = 600 - (140 + 130 + 90 + 60) = 180$
   - $increment = 0 + 49 + 40 + 0 = 89$
   - $a = (0.1 \times 180)/4 = 4.5$

- $b = (0.9 \times 180)/89 = 1.82$
- $NEWB(1) = 0, s = 0$
- $t = a + (b \times 0) + s$
- $NEWB(2) = NEWB(1) + 140 + \lfloor t \rfloor - \lfloor s \rfloor, s = t$
- $t = a + (b \times 49) + s$
- $NEWB(3) = NEWB(2) + 130 + \lfloor t \rfloor - \lfloor s \rfloor, s = t$
- $t = a + (b \times 40) + s$
- $NEWB(4) = NEWB(3) + 90 + \lfloor t \rfloor - \lfloor s \rfloor, s = t$
- $t = a + (b \times 0) + s$
- $NEWB(5) = NEWB(4) + 60 + \lfloor t \rfloor - \lfloor s \rfloor, s = t$

The result is $NEWB(1:5) = (0, 144, 368, 535, 599)$, which is erroneous.

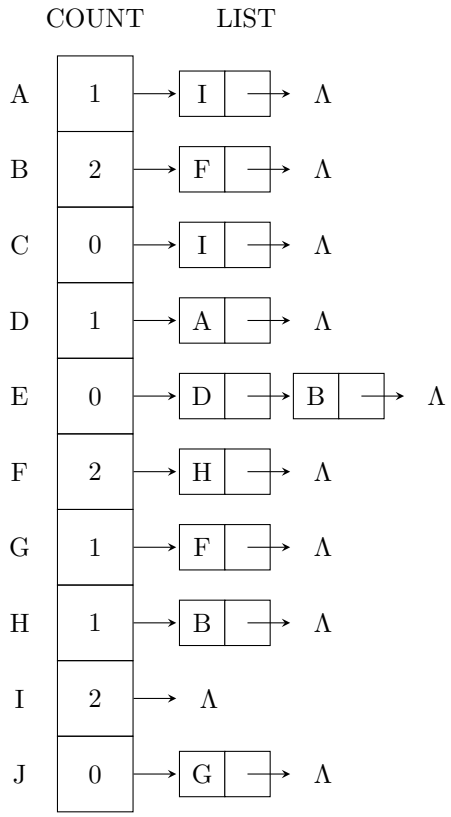
(a) What part/s in the trail of computation contain errors? What should have been the correct computation/s? *(1 point)*
ANSWER:

- 2nd should have been $increment = 0 + 50 + 1 + 40 + 0 = 90$
- 4th should have been $b = (0.9 \times 180)/90 = 1.8$
- 8th should have been $t = a + (b \times 50) + s$

(b) What are the correct new bottom indices of each stack? *(1 point)*
ANSWER: $NEWB(1:5) = (0, 144, 369, 535, 600)$

3. Given the following set of predecessor-successor pair inputs: (G,F), (F,H), (H,B), (C,I), (A,I), (D,A), (E,D), (B,F), (E,B), (J,G). Perform topological sort using the algorithm shown in class.

(a) Show what the adjacency list looks like *before the output* phase (i.e. using queue). *(1 point)*
ANSWER:



(b) Show the sequence of letters generated by the output phase. *(1 point)*
ANSWER: C E J D G A I

Scoring Mechanics

1. For Item 1:
    - **1 point** is given if the (approximate) running time of at least the most costly part and correct time complexity given
    - **0.75 point** is given if the (approximate) running time of at least the most costly part given, but time complexity is erroneous
    - **0.5 point** is given if the time complexity is correct, but justification is erroneous.
    - **0.25 point** is given if the time complexity is erroneous or missing, but significant attempt to justify (albeit also erroneously) was made.
    - **No points** is awarded if no justification was given, even if time complexity was correctly provided.
2. For Item 2a, a **0.25 point *deduction*** is given for each of the following:
    - A missing correct answer from the list
    - An answer given that is not supposed to be in the list.
    - A correctly identified error in the trail of computation, but an erroneous answer as to what should be its correction
3. For Item 2b, a **0.2 point** deduction is given for each erroneous value provided in the new bottom indices.
4. For Item 3a, a **0.1 point** deduction is given for each erroneous feature in the adjacency list.
5. For Item 3b, a **0.2 point** deduction is given for each erroneous letter listed.