Midterm Marking Guide

1

- 1 mark: proper expected value expression
- 1 mark: right answer
- 1 mark: explanation
- 2 marks off for doing the question for only a single value instead of all 3.
- height off-by-one: no marks off.

2

a:

- If you were close to the full definition, 2 marks. Not very close but right idea: 1 mark.

b:

- a very lacking explanation ("Pr<1/m"): 1 mark
- "there's enough space for every element" 2 marks (why does f have this property?)

c:

- saying $n > m \implies \Pr\left(h(i) = h(j)\right) > \frac{1}{m}$ gets 1 mark
- Saying two elements go to the same place: full marks

3

a:

- 1 mark: using more than 3/2 credits
- 2 marks: charge 3 for 0->1, 0 for 1->2
- 1 mark: charging per bit flip (you should charge per increment, which may have multiple tribit flips)

b:

- 1mark: an invariant that says credits will never go negative
- 2 marks: 1 credit stored on twos (no mention of credits stored on ones)

c:

- 0 marks for failing to prove a "never go negative" invariant

- 1 mark off for not proving the case where multiple 2s occur.

4

Many people did not augment the tree with anything other than "count". It was very hard (but not impossible) to implement DEC-BIG-COUNT in $O(\log n)$ if you didn't augment the tree. If you could make it work without augmentation, then you could get full marks.

a:

- *want max*: want a max_count field, that store the maximium count in the subtree rooted at that node

b:

- 2 marks off: *need augmented value:* If you didn't augment the tree, INC-COUNT is trivial to implement (same as insert). You lost 2 marks here unless your DEC-BIG-COUNT implementation actually worked.

- 1 mark off: *update count.*

- 2 marks off: *how do you find v?*

- 2 marks off: *update fields*: don't do anything to update your added fields

- 0 marks for $O(n)$ solutions.

- If you maintain a "weird" added field that doesn't work for part c), you can get up to full marks here.

c:

It was hard to get any marks here if you hadn't augmented the tree with an appropriate field.

- *tree ordered by value, not count:* 0 marks. The tree was ordered by the value field, and so you can't search in the tree according to the count field.

- *this is $O(n)$:* 0 marks. Completely traversing all n nodes in the tree is necessarily $O(n)$.

- *what is v?* 0 marks. There is no node v argument to the function. Read the question carefully.

- *update fields:* 1 mark off. You need to update the max_count field (or whatever field you used) after decrementing.