

HOMework 2 – CS O449

For each implementation style listed: Removal of the last node in a linked list will, assuming it starts from typically known starting points and no caching of any kind, have what runtime performance? (Big O notation using “n” i.e., $O(n)$ or $O(1)$ etc)

Question 1: Single-ended linked list

Answer:

$O(\quad)$

Question 2: Doubly-linked list

Answer:

$O(\quad)$

Question 3: Double-ended linked list

Answer:

$O(\quad)$

Question 4: Double-ended doubly-linked list

Answer:

$O(\quad)$

Question 5:

If you were often appending items to your list in the middle, which data structure would you expect to have the best performance?

A: Linked List

B: Array

Answer:

?

Question 6:

If you are seeing a lot reads but with a random-access pattern while using your list (difficult to predict which item will be accessed next,) which data structure would you expect to have the best performance?

A: Linked List

B: Array

Answer:

?

Question 7:

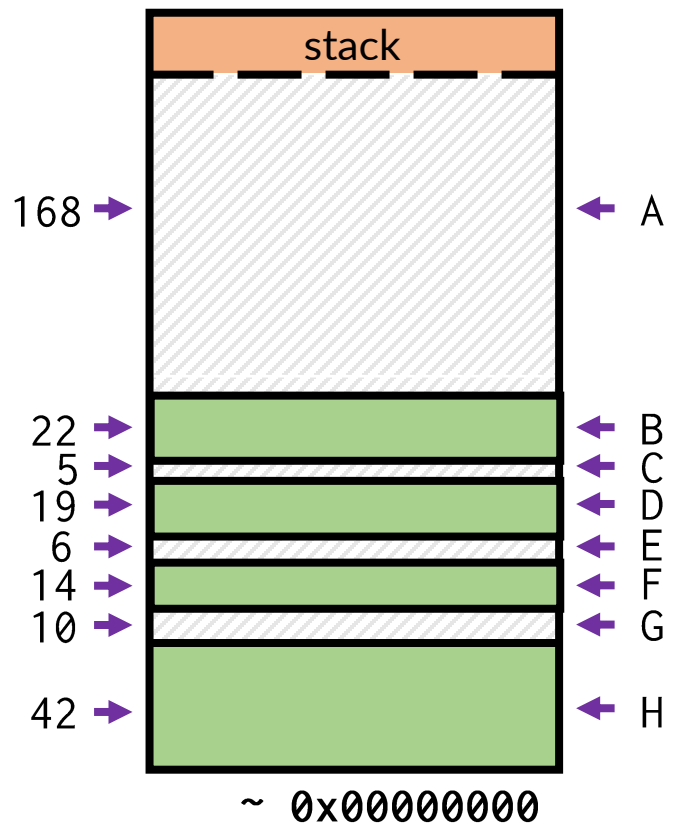
Give a brief explanation about the necessary steps you would have to take in order to perform defragmentation of a program's heap memory. What is a possible problem if you do so carelessly?

Answer:

?

Consider the provided memory layout. The shaded areas are allocated, and the lightly striped areas are available memory. For the given memory allocation of 6 bytes and the given sizes of each region listed to the left of the memory layout graphic, determine which block the allocation will take place in for each listed allocation strategy by placing the letter of that region in your answer box. **Block D was the most recent block allocated.**

malloc(6) → ???



Question 8: Best-Fit

Answer:

?

Question 9: Next-Fit

Answer:

?

Question 10: First-Fit

Answer:

?

Submission:

Please modify this document and answer in the provided spaces and submit your completed document as a PDF to Gradescope. You may write in your answers and scan them in. Or carefully modify this document in Word and export to PDF.