ADSSE Exam June 2022 Answers

Question 1

a) True
b) False
c) True
d) True
e) True
f) True
g) True
h) False
i) False
j) False

1 mark for each correct part

Question 2

a) $\Theta(n^2 \lg n)$
b) $\Theta(\sqrt{\mathrm{n}})$
c) $\Theta(n^4)$
d) $\Theta(n^{0.25} \lg n)$
e) $\Theta(n)$

2 marks for each correct part

Question 3

a) Merge sort: best case $\Theta(n \log n)$; worst-case $\Theta(n \log n)$; no best case or worst case, no constraints on input.
b) Heap sort: best case $\Theta(n \log n)$; worst-case $\Theta(n \log n)$; no best case or worst case, no constraints on input.
c) Quick sort: best case $\Theta(n \log n)$; worst case $\Theta(n^2)$; worst case if input is sorted; best case if chosen pivot is always half way through the array so that each array is partitioned into two equally-sized subarrays.
d) Counting sort: best case and worst case both $\Theta(k + n)$ where input is integers between 0 and *k*.
e) Bucket sort: best case is $\Theta(n)$; worst case is $\Theta(n^2)$ if the contents of each bucket is sorted using insertion sort. Worst case happens when all input items go into the same bucket; best case happens when there is a perfectly uniform distribution of input items over the range.

2 marks for each correct part

Question 4

Needs a description of the randomized select algorithm. Worst-case time is $\Theta(n^2)$.

```
PARTITION(A, p, r)
x = A[r]
i = p − 1
for j = p to r − 1
    if A[j] ≤ x
        i = i + 1
        exchange A[i] with A[j]
exchange A[i + 1] with A[r]
return i + 1
```

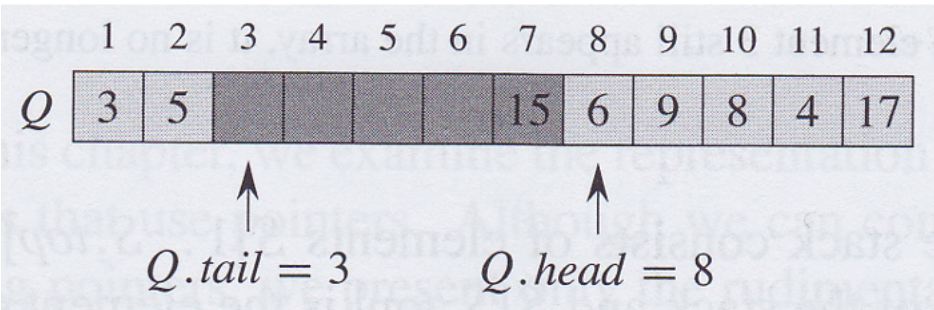
```
RANDOMIZED-PARTITION(A, p, r)
i = RANDOM(p, r)
exchange A[r] with A[i]
return PARTITION(A, p, r)
```

```
RANDOMIZED-SELECT(A, p, r, i)
if p == r
    return A[p]
q = RANDOMIZED-PARTITION(A, p, r)
k = q − p + 1
if i == k            // the pivot value is the answer
    return A[q]
elseif i < k
    return RANDOMIZED-SELECT(A, p, q − 1, i)
else return RANDOMIZED-SELECT(A, q + 1, r, i − k)
```

Question 5

a) Insert, delete, find [3 marks]

b) Stack [1 mark]

c) 

[4 marks]

d) Queue underflow happens when you try to execute a dequeue operation on an empty queue. [1 mark]

e) $\Theta(n)$ [1 mark]

Question 6

a) Load factor is n/m where n is the number of elements in the table and m is the number of slots in the table. [1 mark]

b) Simple uniform hashing occurs if any given input item is equally likely to be hashed to any of the m slots in the hash table. [1 mark]

c) Expected length of a chain is equal to the load factor, n/m. [1 mark]

d) Takes $\Theta(1)$ time to compute hash value and access the appropriate slot in the hash table. Then takes $\Theta(\alpha)$ time to search the chain at that slot for the desired key, giving $\Theta(1+\alpha)$. If *m* is proportional to *n*, then $n = O(m)$ so $\alpha = \frac{n}{m} = \frac{O(m)}{m} = O(1)$. The other condition is that *m* must be proportional to *n*. [3 marks]

e) $O(1)$ time for insertion, also for deletion provided we have found the item that we want to delete. If we haven't found it, then we have to search for it which takes $O(1)$ time on average but O(*n*) time in the worst case. [2 marks]

f) $h(k) = k \bmod m$. If $m = 2^p$, then h(k) will just be the *p* lowest-order bits of *k*, so we are not using all the information in *k* to compute the hash value. [2 marks]

Question 7

a) O(lg n) [1 mark]

b) 2 5 5 6 7 8 [2 mark]

c) $\Theta(n)$ [1 mark]

d) Three times. The algorithm returns a pointer to the node whose key is 8. [2 marks]

e)

```
TREE-MINIMUM(x)
while x.left ≠ NIL
    x = x.left
return x
```

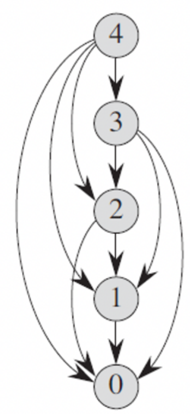
[4 marks]

Question 8

a) Memoized-cut-rod and bottom-up-cut-rod [2 marks]

b) Cut-rod: $O(2^n)$; Memoized-cut-rod and Bottom-up-cut-rod: $O(n^2)$. [3 marks]

c)



[2 marks]

d)

```
EXTENDED-BOTTOM-UP-CUT-ROD(p, n)
1  let r[0..n] and s[0..n] be new arrays
2  r[0] = 0
3  for j = 1 to n
4      q = −∞
5      for i = 1 to j
6          if q < p[i] + r[j − i]
7              q = p[i] + r[j − i]
8              s[j] = i
9      r[j] = q
10 return r and s
```

So needs an extra array, *s*, which stores the length at each step of the new piece to be cut. This is updated in line 8. [3 marks]

Question 9

a) Lines 16-18. [1 mark]
b) Lines 5-10. [1 mark]
c) 3 [1 mark]
d) 1 [1 mark]
e) Line 25. [1 mark]
f) Because the memory allocated on the device cannot be accessed directly from the host and vice versa. [2 marks]
g)
```
 0  1  2  3
 4  5  6  7
 8  9 10 11
12 13 14 15
```
[3 marks]

Question 10

a) Inheritance is where a subclass inherits functionality from a superclass. It is defined at compile time. Developer can override inherited operations. Composition is where a class has a field whose type is the class that has the functionality to be re-used. Composition defined at runtime. [2 marks]
b) Doesn't break encapsulation. If contained type is abstract, then containing type does not depend on implementation of class/interface that provides the desired functionality. [2 marks]
c) The product backlog is the model of work to be done, an ordered list of product requirements, use cases etc. [2 marks]
d) The product owner. [2 marks]
e) The scrum master. [2 marks]