SUNY at Stony Brook — **Midterm 1**
CSE 373 – Analysis of Algorithms — **October 4, 2016**

Midterm Exam

Name: ______________________ Signature: ______________________

ID #: ______________ Circle one: GRAD / UNDERGRAD

INSTRUCTIONS:

- This is a *closed* book, closed mouth exam.
- You must use pencil in the multiple choice component.
- **You must enter the multiple choice on the bubble sheet!!!**
- Check to see that you have 6 exam pages plus this cover.
- Write your algorithms in sufficient detail that we can be sure you understand what you are doing.
- Use the back of the pages if you need more room.
- Look over all problems before starting work.
- Think before you write.
- Good luck!!

Problem	Score	Maximum
1		20
2		20
3		20
MC		$30 \times (4/3) = 40$
Total		100

1) (20 points) Suppose you are given an input set S of n numbers, and a target integer k, and want to find a subset of the elements of S whose sum is exactly k. Further you are given a black box that if given any set of numbers C and integer t instantly and correctly answers true/false whether there is a subset of C whose sum is exactly t.

Show how to use the black box $O(n)$ times to find a subset of numbers of S that adds up to k.

2) (20 points) Consider the task of merging k sorted lists of size n/k. This means that there are $n = n/k \times k$ elements in total. Further, assume that $k \neq \Theta(1)$, i.e. not a constant.

(a) Suppose we merge them by merging two of the lists using the same merge function as merge sort, then merge this list with the next list, and repeat until all n elements are sorted. What is the running time of this strategy? Explain why.

(b) Now suppose we insert all the elements into a heap and repeatedly use extract min to obtain the new array. What is the running time of this strategy? Explain why.

(c) Now suppose we merge each list with another list of the same size, and repeat until we have one sorted list containing all n items. What is the running time of this strategy? Explain why.

3) (20 points)

Suppose we are given a balanced binary search tree data structure of height h, where insert, search, delete, min, max, predecessor, and successor all run in $O(h)$ time.

(a) Show how to use this dictionary data structure to find the kth smallest element in the tree in $O(kh)$ time.

(b) Now assume that each node x in the tree has an extra field (ndescendents) which counts the total number of nodes in the subtree rooted by x.

Show how you can take advantage of this information to find the kth smallest element in the tree in $O(h)$ time.

Multiple Choice

Answer these multiple choice problems in pencil on the bubble sheet! You may write on these pages but they will not be graded.

Mark your test version on the bubble sheet NOW. You have test version **2**.

Neither the TA nor the instructor will answer any questions about these multiple choice problems. Use your best judgement about what they mean.

1. What is the running space of merge sort?

   A. $\Theta(\log n)$ B. $\Theta(n)$ C. $\Theta(n \log n)$ D. $\Theta(n^2)$

2. What is the worst case of heap sort in terms of running time?

   A. $\Theta(\log n)$ B. $\Theta(n)$ C. $\Theta(n \log n)$ D. $\Theta(n^2)$

3. Pick the strongest, correct description of the asymptotic behavior of $f(n)$:.

   $$f(n) = \sqrt{n^2 + 10n}$$

   For the purposes of this question:

   - Θ is stronger than O, which is stronger than Ω
   - A simpler statement is stronger than a more complex one, e.g. $O(n)$ is stronger than $O(n+1)$ or $O(2n)$.
   - For O, $O(g(n))$ is stronger than $O(f(n))$ if $g(n) = O(f(n))$ and $f(n) \neq O(g(n))$. The same is true for Ω

   A. $O(n)$ B. $\Omega(n)$ C. $\Theta(n)$ D. $\Omega(\sqrt{n})$

4. Pick the strongest, correct description of the asymptotic behavior of $f(n)$:.

   $$f(n) = n^3 + n^2$$

   For the purposes of this question:

   - Θ is stronger than O, which is stronger than Ω
   - A simpler statement is stronger than a more complex one, e.g. $O(n)$ is stronger than $O(n+1)$ or $O(2n)$.
   - For O, $O(g(n))$ is stronger than $O(f(n))$ if $g(n) = O(f(n))$ and $f(n) \neq O(g(n))$. The same is true for Ω

   A. $O(n^3)$ B. $\Theta(n^3)$ C. $\Theta(n^3 + n^2)$ D. $\Omega(n^3)$

5. For this question, pick the correct ordering of the given expressions.

   For the purposes of this question, $f(n) < g(n)$ means $f(n) = O(g(n))$, and $f(n) = g(n)$ means $f(n) = O(g(n))$ and $g(n) = O(f(n))$.

   A. $n \log n < n < n^{1+\epsilon} < 2^n < 3^n$
   B. $n < n \log n < n^{1+\epsilon} < 2^n = 3^n$
   C. $n < n \log n < n^{1+\epsilon} < 2^n < 3^n$
   D. $n < n^{1+\epsilon} < n \log n < 2^n < 3^n$

6. For this question, pick the correct ordering of the given expressions.

   For the purposes of this question, $f(n) < g(n)$ means $f(n) = O(g(n))$, and $f(n) = g(n)$ means $f(n) = O(g(n))$ and $g(n) = O(f(n))$.

   A. $\log_{10} n = \log_4 = \log_2 n < n \log n < n^{1+\epsilon}$

   B. $\log_{10} n < \log_4 < \log_2 n < n^{1+\epsilon} < n \log n$

   C. $n^{1+\epsilon} < \log_{10} n < \log_4 < \log_2 n < n \log n$

   D. $\log_{10} n = \log_4 = \log_2 n < n^{1+\epsilon} < n \log n$

7. What is the worst-case running time of the dictionary operation $Search(L, k)$ implemented by an unsorted array?

   A. $O(1)$ B. $O(n)$ C. $O(\log n)$ D. $O(\sqrt{n})$

8. What is the worst-case running time of the dictionary operation $Insert(L, k)$ implemented by a sorted array?

   A. $O(1)$ B. $O(n)$ C. $O(\log n)$ D. $O(\sqrt{n})$

9. What is the worst-case running time of the dictionary operation $Delete(L, k)$ implemented by a doubly linked list?

   A. $O(1)$ B. $O(n)$ C. $O(\log n)$ D. $O(\sqrt{n})$

10. What is the worst-case running time of the dictionary operation $Search(L, k)$ implemented by a sorted doubly linked list?

    A. $O(1)$ B. $O(n)$ C. $O(\log n)$ D. $O(\sqrt{n})$

11. What is the worst-case running time of the dictionary operation $Search(L, k)$ implemented by a sorted array?

    A. $O(1)$ B. $O(n)$ C. $O(\log n)$ D. $O(\sqrt{n})$

12. What is the best-case time complexity of searching in hash table?

    A. $O(\lg n)$ B. $O(n)$ C. $O(1)$ D. $O(n \lg n)$

13. The expected time of finding the minimum value in a min-heap is ( ).

    A. $O(\lg n)$ B. $O(n)$ C. $O(1)$ D. $O(n \lg n)$

14. The worst case time for quick sort is ( ).

    A. $O(\lg n)$ B. $O(n)$ C. $O(n \lg n)$ D. $O(n^2)$

15. Which of the following algorithms *does not* use divide and conquer?

    A. insertion sort B. binary search C. quick sort D. merge sort

16. The worst case time for merge sort is ( ).

    A. $O(\lg n)$ B. $O(n)$ C. $O(n \lg n)$ D. $O(n^2)$

17. How many bits do you need to represent the numbers from 0 to $2^i - 1$?

    A. $i - 1$ B. i C. 2^{i-1} D. 2^i

18. Which algorithm demonstrates the power of using the correct data structure for the job?

    A. Selection Sort B. Insertion Sort C. Heap Sort D. Merge Sort

19. Assume that we are given n pairs, each of the form $(color, number)$ where color is one of $\{red, blue, yellow\}$. We want to sort these pairs first by their color (the order of precedence the same as states) then by their number.

    Now assume that the items arrive already sorted by color. How quickly can we accomplish this task?

    A. $\Theta(n^{3/2})$ B. $\Theta(n \log n)$ C. $\Theta(n)$ D. $\Theta(n^2)$

20. Assume that we are given n pairs, each of the form $(color, number)$ where color is one of $\{red, blue, yellow\}$. We want to sort these pairs first by their color (the order of precedence the same as states) then by their number.

    Now assume that the items in arbitrary (unsorted) order. How quickly can we accomplish this task?

    A. $\Theta(n^{3/2})$ B. $\Theta(n \log n)$ C. $\Theta(n)$ D. $\Theta(n^2)$

21. I've implemented a *min*-heap as an array. What's the time complexity of finding the *maximum* element?

    A. $\Theta(1)$
    B. $\Theta(\log n)$
    C. $\Theta(n)$
    D. $\Theta(n \log n)$

22. What's the running time of mergesort on a doubly-linked list *instead of an array*?

    A. $\Theta(n \log n)$ B. $\Theta(n)$ C. $\Theta(n^2)$ D. $\Theta(2^n)$

23. $f(x) = \Theta(g(x))$ implies that

    A. $f(x) = O(g(x))$
    B. $f(x) = \Omega(g(x))$
    C. All of the above
    D. None of the above

24. What data structure can be used to represent customers waiting on line to pay at the register?

    A. Array B. Stack C. Dictionary D. Queue

25. What data structure *cannot* be used to store an unknown number of data elements?

    A. Static Array B. Dynamic Array C. Singly Linked List D. Doubly Linked List

26. Placing the daily problem in folders according to student name is an example of what data structure?

    A. Array B. Binary Sort Tree C. Priority Queue D. Hash Table

27. Which sorting algorithm is generally more efficient due to less overhead than the others except in its worst case scenario?

    A. Quicksort B. Mergesort C. Heapsort D. Binary Search

28. Given the worst case efficiency of two different algorithms, which runs faster?

    A. $O((1/2)n^3)$ B. $O(n^3)$ C. $O(n^3 + n^2)$ D. Asymptotically equivalent

29. Given the worst case efficiency of two different algorithms, which runs faster?

    A. $O(2^{n+1})$ B. $O(2^n)$ C. Asymptotically equivalent

30. Suppose you need to implement the dictionary using a sorted array of n elements. What is the dictionary operation of search, insert and delete, respectively?

    A. $O(n)$, $O(1)$, $O(1)$
    B. $O(n)$, $O(\log n)$, $O(\log n)$
    C. $O(\log n)$, $O(n)$, $O(n)$
    D. $O(\log n)$, $O(\log n)$, $O(\log n)$