SCC120 Fundamentals of Computer Science

Introduction to Algorithms

Week 8 Workshop

1) The following functions represent the number of statements executed by some algorithms for input data sets of size n. What is the asymptotic (Big O) complexity class (and name) of each function? Rank the functions in terms of order of increasing complexity, that is, worse efficiency.

(a) $f_1(n) = n + 15$

(b) $f_2(n) = 5/4 + 2n^2 + 3n$

(c) $f_3(n) = 10n + n^3 + 234$

(d) $f_4(n) = 432$

2) Consider the code fragment below. Assume the inputs are two arrays of size N and M respectively.

```
for (i = 0; i < N; i++)
    { sequence of
    statements
}
for (j = 0; j < M; j++)
    { sequence of
    statements
}
```

Determine worst-case complexity for each of the following cases. Also explain why.

- Each sequence of statements in each loop is O(1)
- Each sequence of statements has O(N) complexity
- Each sequence of statements in each loop is O(1) but the second loop counts to N instead of M

3) Assume that N is the size of an input array. What is the worst-case complexity of the following code fragment, if "sequence of statements" takes O(1) time? What if the "sequence of statements" take O(N) time?

```
for (i = 0; i < N; i++)
    {  for (j = 0; j < N; j++)
    {
        sequence of statements
    }
}
for (k = 0; k < N; k++)
    { sequence of statements
}
```

4) What is the worst case complexity class of the following code? Assume n is the size of the input array.

```
j = 0;
while (j < n) {
    for (k=0; k<(10*n); k++) {
        ... some O(1) code ...
    }
    j++;
}
```

5) What is the worst case complexity class of the following code? Assume n is the size of the input array.

```
for (j=1; j<n; j=j*10) {
    ... some O(1) code ...
}
```

6) What is the worst complexity class of the following code fragment? Assume the inputs are two arrays of size N and M respectively. In the code, "condition" takes constant time, "sequence1" takes O(N) time, and "sequence2" takes $O(N^2)$ time.

```
for (int i=0; i<M; i++)
    { if (condition) {
        sequence1
    } else {
        sequence2
    }
}
```

What if "condition" takes O(N) time?

7) What is the worst complexity of this code? Assume the input is a table with n rows.

```
for (int i=0; i<n; i++) {
    for (int j=n-5; j<n; j++)
        { for (int k=j; k<n; k++)
        {
            ... some O(1) code ...
        }
    }
}
```

8) When implementing insertion sort using linked lists, a major advantage of linked lists compared to arrays is _____.

A. Faster search speed
B. Smaller storage space
C. Faster insertion operation
D. Faster sorting time

9) In binary search, if the searched element does not exist in the sorted array, the time complexity of the search is _____

10) Binary search relies on the crucial precondition that the array _____.
A. Must be of integer type
B. Must be in contiguous memory space
C. Must be sorted
D. Should not have duplicate elements

11) The advantage of sentinel search over regular linear search is_____.
A. Shorter code
B. Reduced number of comparison operations
C. No additional space requirement
D. More suitable for small-scale data

12) In an ordered list with a length of 97, the maximum number of comparisons needed for binary search is_____.

13)
(a) Give one example of an input (i.e. an array of integers and an integer to search for) where sentinel search is faster than linear search. Please explain the reasons.

(b) Give one example of an input (i.e. an array of integers and an integer to search for) where sentinel search with sorted array (the algorithm in slide titled Sentinel on Sorted) is faster than sentinel search. Please explain the reasons.

(c) Give one example of an input (i.e. an array of integers and an integer to search for) where binary search is faster than sentinel search with sorted array. Please explain the reasons.