

Computer Science 302 Spring 2018 Practice Examination for the Second Examination, March 7, 2018

Name: _____

No books, notes, or scratch paper. Use pen or pencil, any color. Use the rest of this page and the backs of the pages for scratch paper. If you need more scratch paper, it will be provided.

The entire examination is 265 points.

1. True or False. [5 points each]

- (a) _____ $n = O(n^2)$
- (b) _____ $\log(n^2) = O(\log n)$
- (c) _____ Binary search on an ordered list of length n takes $O(\log n)$ time.
- (d) _____ In the worst case, mergesort uses $O(n \log n)$ comparisons to sort n items
- (e) _____ In the worst case, quicksort uses $O(n \log n)$ comparisons to sort n items
- (f) _____ An abstract stack can hold any number of items.

2. Fill in the blanks.

- (a) [5 points] Any comparison-based sorting algorithm on a list of n items uses at least _____ comparisons in the worst case. (Give an asymptotic answer.)
- (b) [5 points] An item on a stack, a queue, or a heap normally represents an _____.
- (c) [10 points] The **only** abstract difference between a stack and a queue is

- (d) [15 points] There are three solutions to the *false overflow problem* for an array implementation of a queue. Briefly, what are they? (Even though I am giving you a whole line for each answer, your answers could be very short.)

3. Using asymptotic notation state how many times “Hello world” will be printed for each of the pseudo-code fragments below, in terms of n .

(a) [0 points]

```
for(int i=1; i<n; i++)  
    cout << "Hello world" << endl;
```

(b) [10 points]

```
for(int i=1; i<n; i++)  
    for(int j=i; j<n; j++)  
        cout << "Hello world" << endl;
```

(c) [10 points]

```
for(int i=1; i<n; i++)  
    for(int j=1; j<i; j=2*j)  
        cout << "Hello world" << endl;
```

(d) [10 points]

```
for(int i=1; i<n; i++)  
    for(int j=i; j<n; j=2*j)  
        cout << "Hello world" << endl;
```

(e) [10 points]

```
for(int i=2; i<n; i=i*i)  
    cout << "Hello world" << endl;
```

4. [10 points] Suppose we are given:

```
void george(int n)  
{  
    if (n > 0)  
    {  
        george(n/2);  
        george(n/2);  
    }  
    else  
        cout << "Hello George" << endl;  
}
```

How many times will “Hello George” be printed if `george(n)`; is executed? Give the answer as an asymptotic function of n .

5. [20 points]

Order the following functions by asymptotic class. Be sure to indicate which functions are in the same asymptotic class. For example, $\frac{n}{6}$ and $5n + \sqrt{n} + 2$ are in the same asymptotic class.

$n, n^2, \log n, n \log n, n \log \log n, \log(n^2), \log^2 n, \sqrt{n}, \frac{(2n+3)^4}{(n+2)^3}, n!, \log(n!), 2^n, n^n, 3^n, 5n + \sqrt{n} + \log n + 15$

6. [20 points]

Using the stack algorithm given in class, evaluate the postfix expression

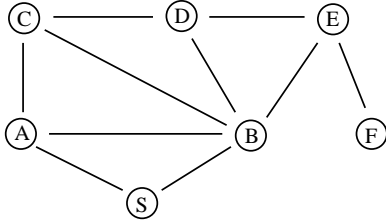
`22*357+3-*+`

showing the stack after each step. Each numerical input is a single digit; for example “22” is not twenty-two, but is rather the number two written twice. However, numbers on the stack can have any size.

7. [30 points] Suppose that the items of a queue Q are C, H, A, T in that order, meaning that C is the front item and T is the rear item.
- (a) Sketch the appearance of a circular linked list implementation of Q , as we did in class (or in any other reasonable style).
 - (b) When you evaluate the function $dequeue(Q)$, what will be the return value of that function?
 - (c) Show the appearance of the circular linked list after evaluating that function.
 - (d) After evaluating that function, execute $enqueue(Q, E)$. Show the steps. (You should draw at least two additional figures.)

8. [30 points] Step through the algorithm that visits all the nodes of this graph in breadth first order, starting with S . At each step, show the queue as well as the output. You can represent the queue informally; for example, as "CBFA" where the front item (C in this case) is on the left. Represent the empty queue by writing ϵ .

Initially, the queue is "S" and the output is empty. At the end, the queue is empty and the output is a list consisting of all the nodes in BFS order.



9. [20 points] Step through polyphase mergesort where the input is the list A,X,Q,B,H,Y,N,C,P,S.

10. [10 points] The following code is for the function “find” in the union/find data structure. Your job is to implement **path compression** by writing one line of code in the correct place.

```
struct vertex
{
    vertex*link;
    // possibly other fields
};

vertex*find(vertex*x)
{
    if(x->link == x)
        return x; // x is a root
    else
    {
        vertex*result = find(x->link);

        return result;
    }
}
```

11. [20 points] **Harder Problems.** Give the asymptotic time complexity for each of the code fragments below, in terms of n .

(a)

```
for(int i=1; i<n; i=2*i)
    for(int j=i; j>1; j=j/2)
        cout << "Hello world" << endl;
```

(b)

```
for(int i=n; i>1; i=log(i)) // assume log(i) truncates to an integer
    cout << "Hello world" << endl;
```