

5. (4 pts) Give the Big-Oh running time of the following:

a) Push back in a doubly linked list with only a head pointer

b) Binary search in an array implemented list

c) Dequeue from an array implemented queue

d) Dequeue from a linked list implemented queue

e) Push back in a singly linked list with a head and tail pointer

f) Remove from the tail of a singly linked list with a head and tail pointer

g) $\lg n + 5n + (n \lg n)/6$

h) $n \lg n + \lg n + n^2 + 1600$

6. (1 pts) Put the following in order of ascending growth rate:

n^n , $500n$, $n \lg n$, $7 \lg n$, $2n^2$, $n^{\lg n}$, 2^n , 4 , $1/n$

7. (3 pts) Given the following snippet of code:

```
int* x = new int;  
int y = *x;  
int* z = &y;  
y = 10;  
int a = *z;  
(*x) ++;  
(*z) --;
```

Describe what the following statements will produce (either what will be printed or what will happen). If a memory address is printed, be as specific as you can by telling me what variable/value the memory address is associated with.

- a) `cout << x << endl;`
- b) `cout << *x << endl;`
- c) `cout << y << endl;`
- d) `cout << &y << endl;`
- e) `cout << z << endl;`
- f) `cout << *z << endl;`
- g) `cout << a << endl;`

8. (4 pts) Write a function that will delete from a singly linked list of integers the node that contains the largest integer. You must accomplish this with 1 pass through the list. Remember to use good programming style. Syntax will be graded. You may assume the following classes exist:

```
class List {  
private:  
    Node* head;  
public:  
    void deleteLargest();  
};
```

```
class Node {  
friend class List;  
private:  
    int item;  
    Node* next;  
};
```