

# Exam 2

## Exam information

CS 250 , Fall 2018

**Name:** \_\_\_\_\_

**Exam Format:** Linked Lists, Stacks, Queues, Algorithm Efficiency, Searching and Sorting

Each question can receive between 0 and 4 points, and each question has a weight associated with it. The point value is used to compute the score for a question. For example, if a question is worth a weight of 5% and the student receives 3 points, then that question will count for 3.75% out of the full 5%.

0	1	2	3	4
Nothing written	Attempted, but incorrect	Partially correct; multiple errors	Mostly correct, one or two errors	Perfect; correct answer & notation

#	Weight	Title	Points Received
1	12%	Stack and Queue	
2	12%	Growth Rates	
3	16%	Diagramming 1	
4	16%	Diagramming 2	
5	12%	Top and Front	
6	12%	Growth Rates	
7	10%	Dynamic Variable	
8	10%	Dynamic Array	
9	+2%	Extra Credit	

## Part 1: Theory

---

### Question 1: Stack and Queue

0  1  2  3  4

When building a Stack and a Queue on top of a Linked List structure, which Linked List functions will be called?

Functions:

PushFront    PushBack    PopFront    PopBack    GetFront    GetBack

Stack's Pop:

Stack's Top:

Stack's Push:

Queue's Pop:

Queue's Front:

Queue's Push:





**Question 6: Growth Rates** 0  1  2  3  4

Identify the growth rates for each function.

a. Growth Rate:  $O(\quad)$ 

```
1 int FunctionA( int n )
2 {
3     int a = 0, b = n;
4     while ( a < n )
5     {
6         while ( b > 0 )
7         {
8             cout << a << ", " << b << endl;
9             b--;
10        }
11        a++;
12    }
13 }
```

b. Growth Rate:  $O(\quad)$ 

```
1 int FunctionB( int n )
2 {
3     int x = 0;
4     for ( int i = 0; i < n; i++ )
5     {
6         x += i;
7     }
8     return x;
9 }
```

c. Growth Rate:  $O(\quad)$ 

```
1 bool FunctionC( int n )
2 {
3     if ( n % 2 == 0 ) return true;
4     else return false;
5 }
```

**Question 7: Dynamic Variable** 0  1  2  3  4

Write a line of code for each of the following:

1. Declare an integer pointer called `ptr` and allocate memory for a dynamic variable via it:
2. Assign the value 100 to the item that `ptr` is pointing to.
3. Display the address that `ptr` is pointing to.
4. Deallocate the memory that `ptr` is pointing to.

---

**Question 8: Dynamic Array** 0  1  2  3  4

Write a line of code for each of the following:

1. Declare a char pointer called `ptr` and allocate memory for a dynamic array of size 3 via it:
2. Assign the values 'A', 'B', and 'C' to the array that `ptr` is pointing to.
3. Display the address of the first element in the array that `ptr` is pointing to.
4. Deallocate the memory that `ptr` is pointing to.

**Question 9: Extra Credit - LinkedList** 0  1  2  3  4

Given these two structures:

```
Node  
string* data  
Node* ptrNext  
Node* ptrPrev
```

```
LinkedList  
Node* ptrFirst  
Node* ptrLast  
int itemCount
```

Implement a function `void Display()` that displays the data of each Node in the Linked List.