CSC326
Name:_____

EXAM 2
CSC326

All work should be done in the blue booklet. This exam, with your name on it, must be handed in with the blue booklet. Good Luck!

1) (10 points) Trace the following code. (You may refer to the header file of the queue that is distributed with this exam.) User input is the following sequence of numbers:

```
4  5  67  89  21  3  0  76
```

```
int main()
{
     QueType<int> numqueue;

        int i;

        cin>>i;

        while (i != 0)
        {
           if (i < 35)
              numqueue.Enqueue(i);
           cin>>i;
        }

        while (!numqueue.IsEmpty())
        {
           numqueue.Dequeue(i);
           cout<<i<<endl;
        }
     }
```

2) Assume an **array implementation** of a STACK that holds integers. (You may refer to the header file for the array implementation of the stack distributed with this exam.) Suppose that you want to count how many positive values and how many negative values are on the stack.

   Given the following client code:

   ```
   StackType IntStack;
   ```

   a) (10 points) Write client code that will print out the count of positive integers on `IntStack` and the count of negative integers. Make sure that when your client code finishes running `IntStack` holds the original values. You may declare any variables you like.

   b) We wish to have a new public member function called `CountPosNeg` that takes two reference parameters. After the call, the first parameter holds the count of

positive integers on the stack and the second parameter holds the count of negative numbers on the stack.

i) (3 points) Show the function prototype and explain where it would be put.

ii) (10 points) Show the implementation code of the function.

iii) (3 points) Give a line of code that shows how the client would use this new member function.

3) Multiple choice (16 points/ 4 points each)

a) Which of the following is true of stacks and queues?

A) A stack is a last-in, first-out structure, and a queue is a first-in, first-out structure
B) A stack is a first-in, first-out structure, and both structures are random access structures.
C) A stack is a last-in, first-out structure, and a queue is a random access structure.
D) A queue is a last-in, first-out structure, and a stack is a first-in, first-out structure.
E) A queue is a first-in, first-out structure, and a stack is a random access structure.

b) Given the code fragment

```
struct NodeType
{
   int  data;
   NodeType* next;
};
NodeType* p;
NodeType* q;
p = new NodeType;
p->data = 12;
p->next = NULL;
q = new NodeType;
q->data = 5;
q->next = p;
```

which of the following expressions has the value NULL?

A) p
B) q
C) q->next
D) q->next->next
E) none of the above

c) Given the declarations

```
struct ListNode
{
   float     volume;
   ListNode* next;
};
ListNode* headPtr;
ListNode* ptr;
float     searchVal;
```

Assume that headPtr is the external pointer to a linked list of many nodes. Which code segment below searches the list for the first occurrence of searchVal, leaving ptr pointing to the node where it was found? (Assume searchVal is definitely in the list.)

A)
```
ptr = headPtr;
while (volume != searchVal)
    ptr = next;
```
B)
```
ptr = headPtr;
while (ptr.volume != searchVal)
    ptr = ptr.next;
```
C)
```
ptr = headPtr;
while (ptr->volume != searchVal)
    ptr++;
```
D)
```
ptr = headPtr;
while (ptr->volume != searchVal)
        ptr = ptr->next;
```
E)
```
ptr = headPtr->volume;
while (ptr != searchVal)
    ptr = ptr->next;
```

d) A C++ class destructor is automatically invoked

A) When a variable is deleted.
B) When one of the data members is a pointer.
C) When the class instance goes out of scope.
D) A class destructor is not called automatically

CSC326
Name:_____

4) (10 points) Complete the following code (where indicated by comments), for the linked list implementation of the stack. (You may refer to the header file for the linked list implementation of the stack distributed with this exam.)

```
void StackType::Push(ItemType newItem)
{
  if( IsFull() )
    cout<<"Fullstack exception thrown";
  else
{
     /* add code here */

}
```

5) Short Answer Questions (28 points):

a) Give a declaration of a dynamically allocated array that holds 500 characters. Declare any variables needed.

b) Why must we be careful to free dynamic memory when we have finished using it?

c) What is the advantage of using the linked list implementation of queues, as opposed to the array implementation?

d) Given the following queue (array implementation), containing the numbers 4, 3, 6, 8 and 9.

	4	3	6	8	9							

i) What are the values of front and rear?

ii) Given this queue, suppose we call Dequeue twice and Enqueue once. What would be the new values of front and rear?

iii) How many elements can this queue hold?

CSC326
Name:_____

6) (10 points) Assume a new member function for the linked list implementation of the unsorted list called `SwitchFront`. `SwitchFront` switches the first elements of two lists. For example, if `list1` contains the elements `a, c, d, r, t` and `list2` contains: `*, &, 6, a, b`, then the call

```
list1.SwitchFront(list2);
```

would leave `list1` with the elements `*, c, d, r, t`
and `list2` with the elements `a, &, 6, a, b`

Write the implementation code for the function `SwitchFront`.