

NAME :

**Open book, open notes, but please do not ask questions.
Write all answers on these sheets.**

question	1	2	3	4	5	total
points						
maximum	20	20	20	20	20	100

1. Write an Abstract Data Type description for complex numbers (numbers of the form $a + bi$, $i = \sqrt{-1}$) where real and imaginary parts can be any number type.
For the operators $*$ and $/$, describe prototype, precondition(s) and postcondition(s).

2. Consider the following code:

```
double A[m][n]; // some m-by-n matrix
double x[n];    // a vector of length n
double y[m];    // will store y = A*x
// statements defining A and x are omitted
for(int i=0; i<m; i++)
{
    y[i] = 0.0;
    for(int j=0; j<n; j++)
        y[i] = y[i] + A[i][j]*x[j];
}
```

(a) Count the number of arithmetical operations in the code above and write the result as a big O statement in the dimensions n and m .

(b) Give a loop invariant on the outer loop controlled by i . Show that satisfying the loop invariant with the negation of the stop condition gives the postcondition.

3. Convert the postfix expression $a\ b\ c\ *\ +\ d\ -\ e\ *$ to an equivalent infix expression. Show the evolution of the stack.

/20

4. Given is a STL deque of type `Job`. Write the definition of a function that takes on input a deque and that returns a copy of the deque as a STL list.
- (a) Every element of the deque should occur in the same order in the list.
 - (b) The code should ***not*** change the given deque!

/20

5. Consider the definition of a node (for any type T) in a double linked list:

```
struct Node
{
    T data; // T is template parameter
    Node *next; // pointer to next node
    Node *prev; // pointer to previous node
    Node(const T& item, Node* next_ptr = NULL, Node* prev_ptr = NULL) :
        data(item), next(next_ptr), prev(prev_ptr) {}
};
```

A (linear) list is stored as a pointer with name `first`. This `first` points to the first item in the list and that item is of type `Node`.

- (a) Draw a list of a least three elements (use `int` as T) and illustrate the steps to remove the first element of the list.

- (b) Write code for the following function:

```
template <typename T>
List<T>::pop_front()
// removes the first element in the list,
// does nothing if the list is empty
{
```