

COSC 350: Quiz 1
Through Section 4.2

Name _____

1. (15 pts.) Which data structure, a *stack*, *queue*, or *deque*, would be the most appropriate for each of the following?
 - a) Holding *activation records* for subroutine calls
 - b) A *LIFO* data structure
 - c) Keeping track of partial paths that need to be searched with a *breadth-first* search strategy
 - d) Simulating a deck of cards where cards are dealt from the top of the deck and all returned cards are added to the bottom of the deck
 - e) Simulating a ticket-line at the theater where people either, a) arrive, enter the line, see it is too long, and leave before anyone else arrives, or, b) arrive, enter the line, and wait until it is their turn to buy a ticket leaving the line once the ticket is bought

2. (6 pts.) Rewrite $(a+b)*(c+d) - e$ in:
 - a) *prefix* notation

 - b) *postfix* notation

3. (6 pts.) Evaluate the following:
 - a) $+ * + 1 2 3 4 * 5$

 - b) $1 2 3 4 * + * 5 +$

4. (4 pts.) Use the *base-conversion* algorithm from the text to convert 111_{10} to base 7.

5. (8 pts.) Rewrite the following code using a recursive function

```
def myIndex(lst, item):  
    at = 0  
    while not item == lst[at]:  
        at = at + 1  
    return at
```

6. (10 pts.) Write a recursive function to double each item on a list of numbers. For example, `double[1,2,3,4,5]` would return `[2,4,6,8,10]`.

7. (18 pts.) $O(3n^2+5n+1)$ in its simplest form is $O(n^2)$. Rewrite each of the following in their simplest form:

a) $O((n^3-n)/n)$

b) $O(5n^2 + n \log n)$

c) $O(2^n+n!)$

d) $O(\log \log n + \log n)$

e) $O(2^n+3^n+4^n)$

f) $O(\log_{10}n + \log_7n + \log_2n)$

8. (12 pts.) If `foo()` is $O(n)$ and `bar()` is $O(n \log n)$, give the complexity for the following chunks of code:

a)

```
ans = 0
for i in range(n):
    for j in range(n)
        ans += i * j
```

b)

```
ans = 0
for j in range(n):
    i = n
    while i > 0:
        ans += i * j
        i = i / 2
```

c)

```
ans = 0
for i in range(n)
    ans += foo(i)
```

d)

```
ans = foo(bar(n))
```

9. In this problem you will develop an implementation for a *Deque*.
- a) (7 pts.) Give an *ADT* for a *Deque*.
 - b) (14 pts.) Give a Python class definition for your *ADT* using Python's built-in list data type.

Pledged: _____