

MIDTERM 1
Introduction to Computer Science COMP 250
Wed. Oct. 5, 2016
Professor Michael Langer

Use the name that corresponds to your McGill email: FIRSTNAME.LASTNAME@mail.mcgill.ca

LAST NAME: _____

FIRST NAME : _____

McGILL ID: _____

GRADE: _____ /10

Instructions:

- This is a closed book exam. You are *not* allowed a crib sheet.
- No electronic devices. You may use your cell phone as a clock only.
- Turn over the exam when you are done (or leave, if you are close to a row exit.)
- Do not talk until *all* the exams are collected.

1. **(1 point)**

What is the following sum ? The numbers are in base 8 and your answer should be in base 8.

$$\begin{array}{r} (143)_8 \\ + \quad (436)_8 \\ \hline \end{array}$$

SOLUTION:

$(601)_8$

We gave 0.5 if you made a careless error but showed that you understood how the carries work and what "mod 8" means.

2. **(1 point)**

Convert the decimal number 85 to binary.

SOLUTION:

$(1010101)_2$

3. (2 points)

Add *two* missing instructions into the pseudocode below.

The method should insert an element *e* into position *i* in an array list *a*[].

You may assume that the index *i* is valid and that the array is not yet full.

Use arrows to indicate where your instructions go.

```
insert(i,e){  
    for (j = size; j > i; j--){  
        <-----  
    }  
    <-----  
    size = size + 1  
}
```

SOLUTION
a[j] = a[j-1]
a[i] = e
[EDIT Oct 13: a[j] = e is also OK, since i==j here.]

Grading scheme:

We gave 1 point for each instruction.

For the first instruction, we gave 0/1 if you got the shift direction backwards and 0.5/1 if you got the direction correct but made an off by one error.

For the second instruction, we gave 0 if you put the instruction inside the **for** loop.

4. (2 points)

Add *two* missing pseudocode instructions below. The method should remove the last element of a *doubly* linked list. Unlike the doubly linked list in the lectures, here you should assume there are *no* dummy nodes.

Use arrows to indicate where your instructions go.

```
removeLast(){  
    tmp = tail  
    <----- tail = tail.prev    //    or tail = tmp.prev  
    <----- tail.next = null  
    return tmp.element  
}
```

Another good solution:

```
tmp = tail  
tail.prev.next = null  
tail = tail.prev  
return tmp.element
```

There was a huge range of answers here. We tried to give points if you got the right idea. Many people had part marks here. I am not going to try to list various answers.

5. (1 point)

Consider a stack. Give a sequence of exactly five **push** and **pop** operations such that:

- the elements are *pushed* onto the stack in the following order: A, B, C, D, E.
- elements are *popped* from the stack in the following order: B, D, E, C, A.

Note that your answer should have five pushes and five pops. You just need to determine the order.

SOLUTION:

push(A), push(B), pop(), push(C), push(D), pop(), push(E), pop(), pop(), pop() Many students included an argument with the pop() instructions which showed which element was popped. We did not take off points for this, but keep in mind that pop does not take an argument.

6. (1 point)

Suppose a circular array of size 4 is used to implement a queue. Show the contents of the array after the following sequence of operations: **enqueue(D)**, **dequeue()**, **enqueue(E)**.

Assume the array state *before* these operations is [- , B, C, -] where B is the head. Note the array indices are [0, 1, 2, 3] .

SOLUTION:

[E , - , C, D]

The main point of the question was to show that you know what a circular queue is, and how it works. Just showing that you know how a queue works is not enough. We did not give any points for that.

7. (2 points)

Use mathematical induction to prove that, for all $n \geq 4$, $2^n \leq n!$.

SOLUTION:

	GRADING POINTS
Base case....	0.5
induction step....	
$2^{\{k+1\}} = 2 * 2^k$	0.5
$< (k + 1) * 2^k$, when $k \geq 4$	0.5
$\leq (k + 1) * k!$, by induction hypothesis	0.5
$\leq (k + 1)!$	

A common error was to start by stating the inequality that you were trying to prove:

$$2^{k+1} \leq (k + 1)!$$

and then write a sequence of inequalities below it. However, without any accompanying explanation, this approach is incorrect, strictly speaking, since it isn't clear which statement implies which, or whether you mean to say that two statements are equivalent.

GRADING SCHEME: For students who made this error, we gave anywhere from 0 to 1 point, depending what was written.