

CS245 Midterm 2

Name:

Start Time:

Finish Time:

I have abided by the Honor Code. I have not discussed this test with anyone.
(Sign below)

If you take this test on separate sheets of paper, make the above your first page.

There are 8 questions in this test. All questions have equal weight. Be sure to answer all of the questions.

1. In the following assignment statements, identify the l-values and r-values and describe all of the work that must be done in Kotlin (in terms of r-values and l-values, etc) for these assignments

```
val a:Int=7  
val b=5  
var c = a + b  
c *= b
```

2. Consider the function definitions for functions written in the imperative style in Kotlin:

```
fun a(bb :Int, cc : Int) : Int
```

```
fun b() : Int
```

```
fun c() : Int
```

Now suppose that somewhere in the program there is a call to the function **a** that looks like

```
a(b(), c())
```

Write the functions **a**, **b** and **c** such that the result returned by the function **a** is different depending on the order of the execution of **b** and **c**

Q 2, Part B: Does/can this issue arise in a program written in the functional programming style. Explain.

3. Suppose you have a programming language in which all mathematical operators have equal precedence. Write an expression that has the same result, regardless of associativity. Write a second expression whose result differs under different associativity. Write a general description of the set of expressions that yield the same result regardless of associativity. For the general description, I am looking for a paragraph (or so) of English text rather than something in Backus-Naur form. (If you are unfamiliar with BNF that is OK since you should not be using it anyway.)

4. One of the design goals of Kotlin is that a program should never (and can never) die with a null pointer exception. What two features of Kotlin work together to achieve this goal? Explain. Examples will help.

5. Kotlin and Go both allow labelling a statement. The break command can then cause program execution to jump to that labelled statement. Describe the restrictions on this use of break (in one of Go or Kotlin). Why do those restrictions exist? Does this use of break run afoul of Dijkstra's critique of GoTo?

6. Following functional programming practices, in Kotlin write a recursive function that takes a list of integers and an input and returns a list of integers such that the last number is the sum of every number in the list, the next to last number is the sum of every number but the first, etc. For instance, given the list (1,2,3,4) the function returns returns (4,7,9,10). Alternately, the function may return this list, reversed (10,9,7,4)
The function should run in $O(n)$ time. If it does not run in $O(n)$ time, describe why.

7. Consider the following Kotlin class definition

```
data class DC(var xx:Int, var yy:String)
```

Write a second class definition — also in Kotlin — that is functionally equivalent to the class

DC. Your new class has the following restrictions

1. It must be a regular class, NOT a “data class”
2. It must NOT have a primary constructor

8. In Kotlin write a command to calculate the sum of the squares of the odd numbers that appear in even indices in a list of integers. For instance given the list (1 2 2 3 3 3 4 4 4 4) the result would be 10 (the only odd numbers at even indices are 1 (index 0) and 3 (index 4)) (If you cannot figure out how to do the whole thing, do a part. Solving piecewise is very much in the functional programming style). Your code must use functions like: map, mapIndexedNotNull, ... It also must the fact that functions are at least second class objects in Kotlin (in fact they are first class).