**THIS IS VERY ROUGH. NO proofing was done.**

**Solutions are in white, or change the solution style.**

1. You can assume a basic Scala coding question on the example. For example, write a printX(), write color class, append to list, arrays, etc..
2. For the following code, what is x and y after the function call is the function is pass by value, pass by reference, pass by value/result, and pass by name? (I’ve asked this question in a exam with a different set of values/function)

func foo(a, b)

a = 2 + b

b = a

b = b + 1

x = 1

y = 2

foo(x, y)

pass by value: 1, 2

pass by reference: 5, 5

pass by value/result: 4, 5

pass by name: 4, 5

1. Assume you have the following (pulled from prior exam).

func f(x, y, z)

x = x + 1

y = z

z = z + 1

a = 1

b = [10,11]

f(a, b[a-1], a) //yes, the the last argument is "a" NOT "b[...]"

What happens when the arguments are ...

**Passed by Value** (place the 3 values separated by a single space):

      What is a,  b[0], and b[1] after f() completes:

(1 10 11, 1, 10, 11)

**Passed by Reference** with no pointer aliasing (place the 3 values separated by a single space):

          What is a, b[0], and b[1] after f() completes:

(3 2 11, 3, 2, 11)

**Passed by Value/Result** (place the 3 values separated by a single space):

      What is a, b[0], and b[1] after f() completes**:**

(2 1 11, 2, 1, 11)

**Passed by Name** (place the 3 values separated by a single space):

          What is a, b[0], and b[1] after f() completes:

(3 10 2, 3, 10, 2)

1. Bottom of Form
2. The classic version of chain of responsibility uses a linked list to store our possible handlers. Why is a linked list preferred over an array in this instance in terms of encapsulation and SOLID principles?

A linked list aids in enforcing order. Array permits random access, which we generally do not want. As a result, we can easily violate D here with an array.

1. Briefly describe 3 important (and significantly different) programming languages developed after 1980. Include application domains, and programming paradigm support. Select languages that we have *not* studied this semester (C++, Java, Python, Scala). Yes, I can ask *very* general questions that overlap with the semester project.

Too many to list!

1. In the below code, what is B() static and dynamic links? Also, how would B() access x? (I’ve asked this question on a exam, with a different set of functions)

Func A()

…

X = 0

Func B()

…

Func C()

B()

C()

Static: A()

Dynamic C()

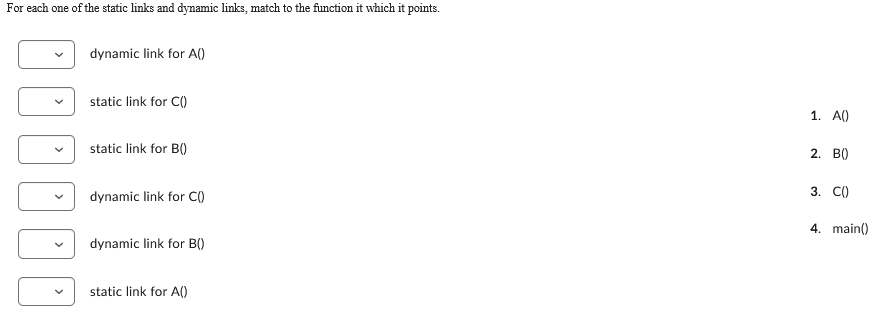
How: uses the static link to see outer function and use its variables if not shadowed

1. For the following problem, consider the following pseudocode (pulled from past exam):

func main()  
 x : integer  
 def A(n : integer)

def B(a : integer)  
 x = a  
 print (a)

def C(a : integer)  
 x = x / 2  
 B(x)

Part 1:

Solution in order: 4,1,1,1,3,4

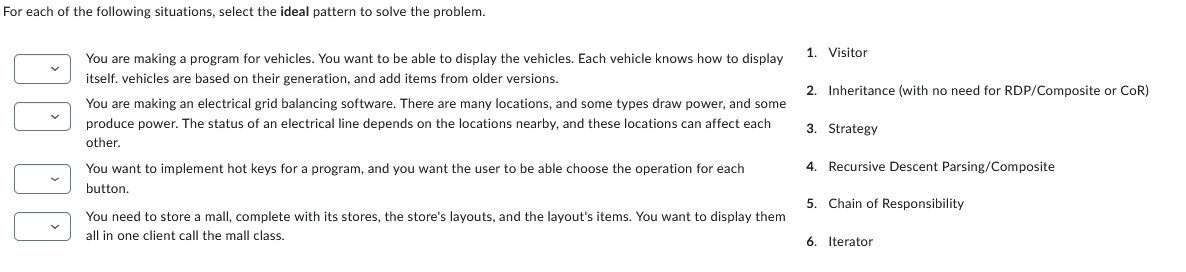
Part 2: Order the function according to their placement on the stack. The first function called should be in the bottom (4th) position.

Solution: B, C, A, main

1. Part 3 Explain how B finds x (2-3 sentences) with static scoping? You must name which links, if any, you are following.

Solution:  It dereferences its static link to find the stack frame of A. Within this frame it finds A’s static link to main within which it finds x.

1. Pulled from prior exam



Solution:

vehicles: best is inheritance, although I took RDP as well because of the tree display

electrical: visitor since a bunch of things are interacting

hot key: strategy, this was one of the examples used in class. You need to assign a function to a key!

mall: best is composite as a mall is a composite entity. I took inheritance as well given you probably want these in one collection.

1. Compare and contrast *functional, scripting, and imperative* programming paradigm? Specifically, what are the key defining features of a language that supports this paradigms?

Functional paradigm is based on math definition of a function: y = f(x1, x2, . . ., xn), and favors immutable data

Imperative is “flowchart” based and explain how to do something rather than what you want.

Scripting favor “ease of use” and glues data and code together. It is almost always dynamical typed.

1. What are side effects, and why do functional languages try to avoid them?

**What**:

**Why**:

Side effects are results that affect object other than those being returned. They make testing far more difficult

1. Compare the built-in-in container data types of Python to those of C++. (I can still ask cross comparison of languages, and since we are at the end of the class, it is the best time to ask this type of question)

C++ has very few built-in container. These include arrays and structs/classes (and unions)..

Arrays: handled by reference, all elements must be of the same type, only supported operation is subscripting.

Struct/class: handled by value, support field access (dot or arrow operator).

Python has a huge number of container datatypes include str, list, tuple, set, dict, range, class, and file. Moreover, they often have built-in support for

* Support for generic functions (len, sorted, sum, min, max, print) and operators (in, +)
* slicing.

Some are immutable (read only): str, tuple.

1. There are 8 main kinds of control structure name 4.(pulled from exam)

* Sequencing
* Selection
* Iterator
* Method/functions
* Recursion
* Concurrency
* Exceptions
* Nondeterminancy

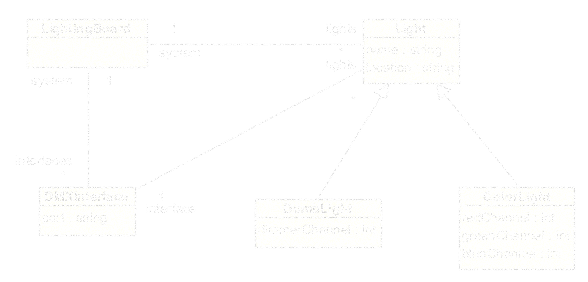
1. Diagram the following problem

**Description**

You are writing the software for a theatrical lighting board. This is a device that controls lights in a theatre. The board has [DMX](http://en.wikipedia.org/wiki/DMX512) interfaces that communicate with the lights. The board connects to lights. There are two types of lights: dumb lights and color lights. Dumb lights have one dimmer channel that controls the brightness of the light. Color lights have three channels for red, green, and blue brightness. Each light is connected to a DMX interface.

**Scenario:** Our board has two DMX interfaces and three lights. One dumb light we are referring to as "lead actor flood" and it uses channel 7 on the DMX controller connected to port "COM1". One color light we are referring to as "door color light" and it is located above the set door. It uses channels 3, 4, and 5 for red, green, and blue on the DMX interface connected to port "COM2". The third light is located at the center-front of the stage and is also a color light.

To view the example, reset the image below.



Explanation:

You are writing the software for a theatrical lighting board.

There's a class right there. It's our system class for this problem. This is a device that controls lights in a theatre.

The board has [DMX](http://en.wikipedia.org/wiki/DMX512) interfaces that communicate with the lights.

DMX interface is a class. It is singular, not plural. There can be many of them. We represent that with an association between LightingBoard and DMXInterface. An interface is associated with one lighting board. A lighting board can have any number of interfaces.

The board connects to lights.

Another class. A light is associated with one lighting board. A lighting board can have many lights.

There are two types of lights: dumb lights and color lights.

There's a dead giveaway that we are going to have inheritance. There are two types, so we have a light base class and two derived classes.

Dumb lights have one dimmer channel that controls the brightness of the light.

I added that as an attribute of DumbLight. From the scenario, I know it is an integer.

Color lights have three channels for red, green, and blue brightness.

These are integers in the ColorLight class.

Each light is connected to a DMX interface.

This implies an assocation between Light and DMXInterface. Each light has one interface. An inteface can be used by multiple lights.

Scenario: Our board has two DMX interfaces and three lights.

This fits our design just fine.

One dumb light we are referring to as "lead actor flood" and it uses channel 7 on the DMX controller connected to port "COM1".

A bunch of additional information here. LIghts need a name (or description). And, the DMX controller needs a port.

One color light we are referring to as "door color light" and it is located above the set door.

More information. Lights need a location.

It uses channels 3, 4, and 5 for red, green, and blue on the DMX interface connected to port "COM2". The third light is located at the center-front of the stage and is also a color light.

This is all taken care of.