Chris Stewart

1. (20 points) Given an arbitrary number of integer arguments, return a list of their squares. For full credit, accomplish this method using map.
squareAll(3, 4, 5) [9, 16, 25]
squareAll(3, 4, 6) [9, 16, 36]
squareAll(1, 3, 5, 5) [1, 9, 25, 25]

```
def squareAll(nums):
    i = 0
    nums2 = []
    for x in nums:
        x = x * 2
        nums2.append(x)

    return nums2
```

2. (20 points) Write a Python class for a simple counter. It should have a property 'count' and a method for incrementing the count by one called increment().

```
class Counter:
    def __init__(self):
        self.__count = 0

    def increment(self):
        self.__count += 1

    @property
    def count(self):
        return self.__count

    def increment_count(self):
        self.__count += 1
```

3. (20 points) Write a short program that asks the user for a name, then for an age, looping until the user types in 'quit' for a name. Then the program should print out the last ages entered for each name. Use the language from your unfamiliar language project. Those using prolog and scheme for their projects may instead complete this problem in Python, but also must include a text answer explaining in detail the differences with their language that would make this impossible, or how it might be accomplished with their language.

```
package main

import (
   "fmt"
   "strings"
)

func main() {
   var name string
   var age float64
   var boolean bool
```

```
    fmt.Println("Enter a name:")
    _, _ = fmt.Scanf("%s", &name)
    fmt.Println("Enter an age:")
    _, _ = fmt.Scanf("%f", &age)
    fmt.Printf("%s is %.0f years old\n", name, age)
    boolean = strings.Contains(name, "quit")

    for boolean == false{
    fmt.Println("Enter a name:")
    _, _ = fmt.Scanf("%s", &name)
    fmt.Println("Enter an age:")
    _, _ = fmt.Scanf("%f", &age)

    fmt.Printf("%s is %.0f years old\n", name, age)

    }
}
```

4. (20 points) Would switching CSC184 to Python accomplish the same goals of switching it to less-java? Give a complete and thorough response leveraging evidence both from the paper and your understanding of Java and Python, about 5-8 sentences in length.

I personally do not believe it would, because they are very different languages. Python is fast and syntactically simple while less-java is just a simpler form of java. Switching to python in my opinion would be more beneficial to students that anything. Python is a simple fast language that give the user the access to a full language with a simple syntax and type inference. A class taught in Python would be capable of so much more than a class taught in less-java and would be easier on new students who might get overwhelmed with a more complex syntax. In recent years it has became less likely a programmer will need to know java and more likely employers are looking for Python experience.

5. (20 points) You've been hired by Initech due to your polyglot (multiple languages) programming skills. Two years ago, Initech moved to an automated system for their TPS reports developed by Intertrode, but are not happy with the system. They want to switch to a new system using MongoDB, but there is no software that can do the database migration automatically. They've asked you to develop some code that will handle the migration for them. They are aware that you are proficient in both Java and Python, so they've asked you to write a short 5-8 sentence explanation of which of the two languages you plan to use and what your justification is. This is an expensive project, so they want this choice to be well justified, including specifics. Please provide your response below.

In my opinion is that Python would be a superior to Java in this case for its speed and the fact that it is a scripting language. Python has type inference and when dealing with large data sets this can be very useful. Scripting languages are built for this type of thing specifically. Scripting languages are made for fast code made for automation. A good example of this was our logic programming assignment. This assignment dealt with a large amount of outcomes and python handled them easily. Java is not good for

this because of its garbage collection this is an expensive was of dealing with data and would greatly drive up the cost of the project.