

**San José State University**  
**Computer Science Department**  
**CS152, Programming Paradigms, Springs 2022**

**Homework #5**

## Objective:

This homework's objective is to practice programming in Scheme programming language.

## Details:

Solutions to all exercises for this homework assignment should be implemented in Scheme. Submit solution to each exercise in a separate .scm file. Make sure to include the following comments into each .scm file (replace with your information):

```
; CS152 Spring 2022  
; Student name  
; SJSU student ID  
; Homework assignment 5  
; Exercise X
```

Name each .scm file with the exercise number. For example, exercise1.scm, exercise2.scm, etc.

### Exercise 1:

Implement a function named *linear* that describes a line given values of m (line slope) and b (intercept):

$$y = m * x + b$$

Implement a function named *quadratic* that describes a quadratic curve given values of a, b, c:

$$y = a * x^2 + b * x + c$$

Implement a third function named *chooseMapping*, which takes in two parameters, one is a string that specifies the transformation, and the second one is a list. If the transformation is “linear”, then *linear* function is applied to the list, with m = 5 and b = 3. If the transformation is “quadratic”, then *quadratic* function is applied to the list, with a = 3, b = 4, and c = 1. Use map function to apply the appropriate function to the input list. You might want to implement some helper functions not specified in this description.

## Homework # 5

Example of your code execution:

```
(display (chooseMapping "linear" '(1 2 3 4 5 6 7 8 9 10)))
```

Expected output:

```
(8 13 18 23 28 33 38 43 48 53)
```

Example of your code execution:

```
(display (chooseMapping "quadratic" '(1 2 3 4 5 6 7 8 9 10)))
```

Expected output:

```
(8 21 40 65 96 133 176 225 280 341)
```

### Exercise 2:

Implement a function named *countOccurrences*, which given an element and a list of elements counts the occurrence of that one element in the list. If the element is not present, then 0 should be returned.

Example of your code execution:

```
(display (countOccurrences 2 '(1 2 1 4 2 6 7 2 1 2)))
```

Expected output:

```
4
```

Example of your code execution:

```
(display (countOccurrences 1 '(1 2 1 4 2 6 7 2 1 2)))
```

Expected output:

```
3
```

Example of your code execution:

```
(display (countOccurrences 10 '(1 2 1 4 2 6 7 2 1 2)))
```

Expected output:

```
0
```

### Exercise 3:

In this exercise you are expected to use higher-order functions and recursion. Implement a function named *reverse*, which given a list reverses the order of the elements in that list.

Example of your code execution:

```
(display (reverse '(1 2 3 4 5 6 7 8 9 10)))
```

Expected output:

```
(10 9 8 7 6 5 4 3 2 1)
```

### Exercise 4:

## Homework # 5

Implement a function named `getNthElement`, which takes in two parameters: a list of elements, a position in the list to retrieve. If the position is out of bounds, then your function should return `False` (`#f`), otherwise the element at the corresponding position should be returned. Indexing should start at 1, not 0.

Example of your code execution:

```
(display (getNthElement '(0 1 2 3 4 5 6 7 8 9 10) 40))
```

Expected output:

```
#f
```

Example of your code execution:

```
(display (getNthElement '(0 1 2 3 4 5 6 7 8 9 10) 0))
```

Expected output:

```
#f
```

Example of your code execution:

```
(display (getNthElement '(0 1 2 3 4 5 6 7 8 9 10) 4))
```

Expected output:

```
3
```

## Submission:

Compress all the homework `.scm` files into a single compressed file named “*YourNameAssignment5*”, with the appropriate file extension. Make sure submit by 11:59pm on the due date listed in Canvas. Submit your solution via Canvas.

If you have any questions, message me or the grader or both:

[Yulia.Newton@sjsu.edu](mailto:Yulia.Newton@sjsu.edu)

[madhujitaranjit.ambaskar@sjsu.edu](mailto:madhujitaranjit.ambaskar@sjsu.edu)

## Grading:

I will return the grades as fast as we can grade this homework. Normally it should not take more than a few weeks.

A total of 20 points are possible for this homework assignment.