

# CSE 3302/5307 Lab Assignment 1

Due June 26, 2013

## Goal:

Understanding of Scheme and elementary functional programming concepts.

## Requirements:

1. Write a Scheme function to implement a general `project` function to extract elements of a list and combine them together to produce an s-exp. The format of an application of the `project` function will be:

```
(project project-exp g-exp)
```

where `project-exp` is an s-exp whose atoms are positive integers and `g-exp` is a list whose atoms are unrestricted.

The result of a `project` takes the `project-exp` and substitutes for each of its atoms the sub-expression at that relative “ordinal” position from `g-exp`.

If an error is detected, then use a string atom of form `"error: ..."` within the output list.

2. Email your program to `jing.xu@mavs.uta.edu` by 10:15 a.m. on June 26, 2013.

## Getting Started:

1. Don't be concerned about efficiency.
2. Some examples:
  - a. `(project '(3 1 2) '(a b c))` should return `'(c a b)`.
  - b. `(project '(3 1 (2 1 (3))) '(a (b c) d))` should return `'(d a ((b c) a (d)))`.
  - c. `(project 2 '(a (b c) d))` should return `'(b c)`.
  - d. `(project 3 '(a (b c) d))` should return `d`.
3. Assuming no errors, `(project 3 g-exp)` produces the same result as `(car (cdr (cdr g-exp)))`
4. The Ten Commandments and The Five Rules from *The Little Schemer* will lead you to many days of happiness.
5. `set!` will lead to nights of suffering (and loss of points).