

Laboratory Assignment 6

Objectives

- Work with pairs and lists

Activities

1. Define a SCHEME function to take a number, and return a pair with the number and its square.
2. Define a SCHEME function `square-1` which will take any list of numbers as input, and returns a list with each of those numbers squared.
3. Define a SCHEME function `range` that takes a pair (x, y) of natural numbers as input and returns a list of all integers between them, inclusive.

For example if $p = (0, 10)$ then

$$(\text{range } p) \rightarrow' (0\ 1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10)$$

4. **Scalar-Vector Multiplication** Multiplying a vector by a scalar produces a vector where each component is the corresponding value in the original vector multiplied by the scalar quantity. For example:

$$a(x_1\ x_2\ x_3) = (ax_1\ ax_2\ ax_3)$$

Define a SCHEME function, named `sv-mult`, which takes a list and a value as parameters and performs scalar-vector multiplication on them.

5. **Vector Addition** Adding two vectors produces a vector where each component is the sum of the corresponding values in the original vectors. For example:

$$(x_1\ x_2\ x_3) + (y_1\ y_2\ y_3) = (x_1 + y_1\ x_2 + y_2\ x_3 + y_3)$$

Define a SCHEME function, named `v-add`, which takes two lists and performs vector addition on them. *Note: **vector subtraction** is structured in the same way.*

6. The **dot product** of two lists of numbers $(x_1\ x_2\ x_3)$ and $(y_1\ y_2\ y_3)$ is

$$x_1 * y_1 + x_2 * y_2 + x_3 * y_3$$

Define a recursive SCHEME function `(dot x y)` that takes two lists of numbers as its inputs and returns the dot product of those two lists. Do not use the in-built `map` function. You can assume the two lists have the same length.

7. The **cross product** of two sets represented by lists of numbers $X = (x_1 \ x_2 \ x_3)$ and $Y = (y_1 \ y_2 \ y_3)$, denoted $X \times Y$, is the set (list) of all possible pairs of numbers where the first number in each pair is a member of the first set (list) and the second number in each pair is a member of the second set (list). In our example, $X \times Y$ is $((x_1 . y_1) (x_1 . y_2) (x_1 . y_3) (x_2 . y_1) (x_2 . y_2) (x_2 . y_3) (x_3 . y_1) (x_3 . y_2) (x_3 . y_3))$. Define a SCHEME function `(cross x y)` which takes two lists as parameters and returns the list of pairs representing the cross product of the items in the lists `x` and `y`.