# Homework Assignment 4

**Any automatically graded answer may be manually graded by the instructor.** Submissions are expected to only use functions taught in the course. If a submission uses a disallowed function, that exercise can get zero points. Excluding promises, *all functions that mutate values are disallowed* (mutable functions usually have a `!` in their name).

## $\lambda$-Racket

*Note: This section must use the AST defined in file* `hw4-util.rkt` *whose functions are prefixed with* `r:`.

1. (30 points) *Your goal is to implement the substitution operation, notation* $e[x \mapsto v]$. Implement function `(r:subst exp var val)` where `exp` is an expression `r:expression?`, `var` is a variable `r:variable?`, and `val` is a value `r:value?`. Function `r:subst` must return an expression of type `r:expression?`. Test cases are included in the template file.

2. (30 points) *Your goal is to implement the evaluation of expressions using substitution, notation* $e \Downarrow v$. Implement function `(r:eval subst exp)`, where `subst` is a variable substitution function given by the system,[1] and `exp` is an expression of type `r:expression?`. Function `r:eval` must return a value of type `r:value?`. Test cases are included in the template file.

## $\lambda$-Racket with environments

*Note: This section must use the AST defined in file* `hw4-util.rkt` *whose functions are prefixed with* `s:`.

3. (25 points) *Your goal is to implement the evaluation of expressions using environments, notation* $e \Downarrow_E v$. Implement function `(s:eval env exp)` where `env` is a hash-table of type `hash?`, whose keys have a type `s:variable?` and values have a type `s:value?`, and expression `exp` has type `s:expression?`. Function `s:eval` must return a value of type `s:value?`. Test cases are included in the template file.

## Manually graded questions

4. (7.5 points) **Manually graded.** Describe one situation where implementing $\lambda$-Racket without environments is a better alternative than $\lambda$-Racket with environments. Conversely, describe one situation where $\lambda$-Racket with environments is a better alternative than $\lambda$-Racket without environments.

5. (7.5 points) **Manually graded.** Describe two benefits of using a formal specification to help with the implementation of a software system.

[1] We choose to make variable substitution a parameter of evaluation so that Exercise 2 can be graded independently from Exercise 1.