Guideline

Homework for Week 04

Please submit your solutions as a single PDF on Brightspace.

Remember that in lambda calculus, a **Term** can take one of the following three forms:

- A **variable** (e.g., `x`, `y`).
- A **function definition**, which is written as `λ Var . Term`.
- A **function application**, which is written as `Term Term`.

We will use this terminology throughout the exercises to describe valid constructs.

Exercise 1: Conditional Function (25 points)

Write a function that takes two arguments. The first argument can be any term, and the second argument will always be a Church boolean. The function should return the first argument if the second argument is `true`. If the second argument is `false`, the function should return the identity function `λx. x`. To solve this, you are only allowed to use variables, function definitions, and function applications. No additional constructs or shortcuts should be used.

Exercise 2: XOR on Church Booleans (25 points)

Write a function that implements XOR (exclusive-or) for Church booleans. XOR is true if exactly one of the inputs is true, and false otherwise. In this exercise, you must build XOR from scratch using only variables, function definitions, and function applications. Do not use any predefined boolean operators such as `AND`, `OR`, or `NOT`.

Exercise 3: Multiplication for Church Numerals (25 points)

Write a function that takes two Church numerals, `n` and `m`, and computes their product `n * m`. Your implementation must use only variables, function definitions, and function applications. You are not allowed to use predefined multiplication operators.

Exercise 4: Square a Church Numeral (25 points)

Write a function that takes a Church numeral `n` and returns its square, `n^2`. Like the previous exercises, you are restricted to using only variables, function definitions, and function applications. You may not use multiplication or other arithmetic operators directly.