1. Simple type inference

    (1) int -> int -> int -> int
    (2) (t_a,int) -> (int,int)
    (3) Sum: (int -> int) -> int -> int
    (4) t_1 -> t_2
    (5) int -> bool

2. Hindley Milner
    (1) Cannot be typed in simple type inference as x is bool in the if condition, and x is int in the if-else branches. So they cannot be unified.
    It cannot be typed in Hindley Milner too as there is no let construct on x.
    (2) Simple type inference:
    (t1 -> t1 -> t2) -> t1 -> t2
    (3) Cannot be typed in simple type inference as (g x) is bool in the if condition and in the first branch of if, but in the else branch 1 + (g y) makes it int. So there is a contradiction.
    It cannot be typed in Hindley Milner too as there is no let construct on (g x).
    (4) Cannot be typed in simple type inference as arguments to h have type Bool and (t1,t1).
    It cannot be typed in Hindley Milner too as g comes from outer scope without let and cannot be polymorphic.
    (5) Simple type inference of herbert y:
    Bool -> int