# Assignment #5

Name: _____  ID: _____

This assignment has **3** questions, for a total of **25** marks.

Question 1: ........................................................................................... 5 marks
Write the proof for the preservation theorem for the cases related to recursive types.

Question 2: **Z combinator typing** . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5 marks

This is the Z combinator in ULC:

$$\lambda f.\,(\lambda x.\,f(\lambda y.\,((x\ x)\ y)))(\lambda x.\,f(\lambda y.\,((x\ x)\ y)))$$

Add type annotations as well as fold/unfolds and prove it can be typed in System F + isorecursive types. Its type is $((\tau_1 \to \tau_2) \to (\tau_1 \to \tau_2)) \to (\tau_1 \to \tau_2)$ for arbitrary $\tau_1$ and $\tau_2$.

Question 3: **Encoding ULC into System F using recursive types** .......................... 15 marks

Try to define type $\tau_u$, which is the type that any ULC term can be given in F+isorecursive types. If you can define $\tau_u$, define a function that maps ULC terms to terms of F+isorecursive types of type $\tau_u$ that preserves behaviour (i.e., if you map an ULC application, you get something that behaves like an application). If you cannot define $\tau_u$, argue why it cannot exist.