Assignment #2

Name: _____ ID: _____

This assignment has **1** questions, for a total of **25** marks.

Question 1: **Fully Abstract Compilation with Pairs** ..... 25 marks

Add pairs ($\langle e, e \rangle$) and projections ($e.1$, $e.2$) to both languages, add their static (typing) and dynamic (operational) semantics to both languages. Define a compiler from the two languages.

In order argue whether it's fully abstract, you'll need to define a backtranslation, so you need to define the backtranslation type. What is the backtranslation type when the target language has **booleans**, **natural numbers** and **pairs**?

Hint: the Cantor Paring Function lets one encode a pair of natural numbers into one (https://en.wikipedia.org/wiki/Pairing_function).

Once you have defined the backtranslation type, define the inject-extract functions, argue why they are correct, define the whole backtranslation and argue whether the compiler is fully abstract. Note that I wrote 'argue' and not 'prove', but be as precise as you can.

If you think there is no encoding that supports this backtranslation, argue why it is the case and why can fully abstract compilation not be attained.

The text below is not a graded exercise. Now, pairs have fixed size, dictated by their type and thus the compiler uses use static information to write the dynamic typechecks. For your own amusement[1], think about how would these typechecks be done with arrays (whose type does not mention length but whose length is fixed) and with vectors (whose type does not mention length and whose length is dynamic). What kind of operations on these datatype would you need?

[1] Do it anyway, there's an oral exam after all.