1. 
   a. Name and define each of the type systems available. (4 points)

   b. give a major programmatic advantage **for each** typing system available. (6 points)

   c. give a major programmatic disadvantage **for each** typing system available. (6 points)

   d. Name the *category* of errors that should be caught by a typing system. (2 points)

   e. Give two types of errors in this *category*. (4 points)

   f. Name the *kind* of languages that guarantee that such error are caught. (2 points)

   g. Name two languages which guarantee to catch such errors. (2 points)

2. 
   a. State the best way to deal with failure in programming. (2 points)

   b. Give a plus and a minus for checked exceptions in Java. (4 points)
3.  
   a. State the differences between the following types: array, structs (or records), and sum of types. (8 points)

   b. Give a programming use of each of those types. (6 points)

4.  
   a. List all array attributes. (6 points)
5.  
   a. Define Name equivalence. (2 points)

   b. Define structural equivalence. (2 points)

   c. Which type equivalence allows you to add oranges to apples and why? (4 points)

6. List all composite types a programming language should offer minimally. (6 points)

7.  
   a. Java’s (before 1.5) typing system is qualified as invariant; explain what this means (4 points)

   b. Provide a programming fragment that illustrates a negative impact by having the typing system qualified as such. (4 points)
8.
   a. What is genericity. (2 points)

   b. what 2 programming abstractions should allow for a generic version. (2 points)

   c. how does C, Java, Ada, Haskell (choose to languages) answer to the desire for the 2 generic constructs? (8 points)

   d. How is genericity handled in your programing language of study. (Name the language). (6 points)

9.
   a. List purposes of a module construct in PLs. (4 points)

   b. Java does not have modules per se, but classes and packages serve that role. Give trade-off of such choice. (4 points)