Warning: This homework is to be done by you and you alone.

For this homework choose one language of the ones listed below to answer questions on it.
Points: 150 pts. And based on quality of answers you may get up to 50 extra bonus points.

- Scheme: http://www.cs.rice.edu/CS/PLT/
- Perl: http://www.perl.com/CPAN/ports/
- Python: http://www.python.org
- Smalltalk: http://www.squeak.org
- Ruby: http://www.ruby-lang.org
- Self: http://dmoz.org/Computers/Programming/Languages/Smalltalk/Self/
- Dylan: http://www.double.co.nz/dylan/introduction.htm
- Tk1-Tk: http://www.tcl.tk/
- Rexx: http://www2.hursley.ibm.com/rexx/

The questions below apply to the programming language chosen for study. For the writing of the answers:

- i. clearly repeat the question.
- ii. provide the answer.
- iii. Provide references indicating sources for the answer.

1. a. Language name: __________________________
   b. Date of first release of the language.
   c. Name of designer or designers.
   d. Is the actual language version you are using the original version? an extension of it? a subset of it? neither an extension nor a subset but a version based on the original language? Explain your answer.
   e. If not the original version, state the date of the release of the version you are using, as well as the date of the release of the translator (compiler, interpreter) you are using to write programs.
   f. Does the language have an standard? if so, state release date, and state whether the translator you are using implements the standard.
   g. If the language does not have a standard, list the documents you are using for the release.

2. a. Describe the component(s) for the formation of a complete program.
   b. List all the compilation units.

3. Give 2 program examples:
   a. One that you can steal. Produce an example of its execution. State clearly the input used in the execution and the output produced.
   b. One that is "representative" of the language application area and illustrates the flexibility of the paradigm.

Note: You can steal each one but you must be able to read, understand and explain. I will choose X number of people at random to come to my office for this part of the homework.

Important: for all programs you must submit a complete example of input, execution and output.
Warning: This homework is to be done by you and you alone.

4. a. Describe kinds of blocks provided. Can they be nested?
b. What kind of scope discipline is provided by the language.
c. If it provides more than one mechanism how can you specify which one to use?

5. Clearly and completely describe the following issues about its type system:
   - primitive built-in types.
   - composite built-in types.
   - user-defined composite types
   - methods expected to be used by programmer to identify type of actual values.
   - implicit/explicit conversions
   - escapes from the type system.
   - Data structure provided to deal with homogeneous collection of values.
   - Data structure provided to deal with heterogeneous collection of values.
   - Data structure provided to deal with list of heterogeneous values.
   - What are consider language’s values and of those which ones are first-class values.

6. Parameter passing mechanisms:
   - What type of parameter passing mechanisms your language provide. Name each and give an example of its use.
   - Can you specify functions as parameters?
   - Can you return functions as values?
   - Can you specify function types and declare variables of such types? Provide example if so.

7. Generics:
   - Can you specify generic data structures? If so is the parameter of the generic definition a type or is it something else?
   - Can you specify generic methods? Provide an example of such if so.

8. Control structures:
   - List all the forms of conditionals and loops provide by the language. Give an example of each. If the language does not provide a mechanism for loops, explain the construct that replace them and provide an example.

9. Exceptions:
   - Can exceptions be declared in the language. If so provide an example, and provide an example where exceptions are raised and handled.
   - If no exceptions are provided by the language, how are errors transmitted and handled by programmers.

10. a. What kind of scope disciplines are provided by the language.
    b. If it provides more than one mechanism how can you specify which one to use?

11. a. Can you specify statically allocated memory? how? provide a programming example.
    b. On management of dynamic memory:
       i. How is allocated? how is deallocated?
ii. Does the language documents talk about a garbage collector? is it expected by an implementation?

iii. If the language provides a garbage collector, are there statements in the language that the programmer can use to direct the garbage collector to collect some garbage explicitly? and if directed will it do so?

12. Describe the language mechanisms (modules or other) the language provides for the organization of program components to be used for writing of programs in teams

13. a. What kind of scope disciplines are provided by the language.

b. If it provides more than one mechanism how can you specify which one to use?

14. For graduate students only.
For both languages, the language you chose and Haskell, write a separate report discussing the features the language provides for concurrent programming. The report should be at most two pages long for each language. Note: most modern languages provide extensions with concurrent mechanisms.