Paper on chosen language, Part II

All question below are w.r.t. the language chosen. You must give exact references of the documents used to answer the questions giving exact locations in them.

0. Name of language.

1. If the language chosen only provides for interpreters, give language reasons for such restriction. Illustrate them with language statements that can only be implemented by interpreters.

2. 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?

3. a. What storage methods does the language use? For each method it uses provide a fragment of code illustrating use the method.
   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?

4. a. What kind of type system method the language provide? static, dynamic, latent?
   b. Is the language strongly typed?
   c. Give an example of a typing error that are only caught at run-time. And give an example of an error caught at run-time that could have been caught at compile time. If you cannot provide any of the last kind, explain why.
   d. Does the typing system allow for escape hatches? Clearly illustrate them with code fragments.
   e. What kinds of values does the language support? Classify the possible kinds of values into first-class or second class. Explain when a kind of value is second class.

5. a. Provide the mechanisms the language has for user-defined data types. Clearly distinguish between mechanism to define new scalar types and mechanisms to define composite types.
   b. Are you able to add apples to pears?
   c. What kind of type equivalence the language uses?
   d. Describe the kinds of type transfers between values the language provides carried out either implicitly by the language or explicitly by the programmer. Illustrate them with code examples.
   e. Can you transfer a value from a larger type to a smaller type?

6. a. Specify the syntax and give examples for:
   i. Definition of constants for primitive types as well as user defined types.
   ii. Specification of literal values for user defined types, including composite types.

   b. What is the semantics of variables. What is the meaning of assignment in the language.
c. Provide all the forms of expression formation given by the language. In particular this includes the specification of all operators available by the language as well as means of defining new operators.

7. Describe the language mechanisms the language provides for the organization of program components.