1. For the **procedural** programming paradigm: Answer the following:

   a. Programming language using such paradigm.___________________ (1 points )
   b. model of computation used by paradigm:(2 points )
   
   c. Programming methodology to be used with the paradigm.(3 points )
   
   d. List factors than enhance and factors that hinder program reliability using such paradigm.(4 points )

2. For the **functional** programming paradigm answer the following questions:

   a. Programming language using such paradigm.___________________ (1 points )
   b. model of computation used by paradigm:(2 points )
   
   c. Programming methodology to be used with the paradigm.(3 points )
   
   d. List factors than enhance and factors that hinder program reliability using such paradigm.(4 points )
3. Consider the following factors dealing with expressions: form of writing of expressions, evaluation time of expressions, evaluation order of expressions. For each below, name each factor that impacts it, positively or negatively, explaining why:
   a. readability (2 points)
   b. writeability (2 points)
   c. program efficiency (2 points)

4. a. For the production of correct programs, which attribute of a variable, different from its name, is benefited by having the earliest binding; name a language that supports your answer. (3 points)
   b. For the efficient writing of programs, which attribute of a variable, different from its name, is benefited by having the latest binding; name a language that supports your answer. (3 points)
   c. For the use of functions, which attribute different from its name, is benefited by having the latest binding time; name a language that supports your answer. (3 points)
5. a. Define the two semantical meanings found for variables, and discuss their trade-off. (6 points)

b. Explain the semantics of variables in Java. (2 points)

6. Give a complete piece of code in your favorite language illustrating all the memory allocation methods. (6 points)

7. For each of the statements below, if true give an example of language supporting it; if false explain.
   a. Would it make sense to have a PL with static scope and static typing? (2 points)

   b. Would it make sense to have a PL with static scope and dynamic typing? (2 points)

   c. Would it make sense to have a PL with dynamic scope and dynamic typing? (2 points)

   d. Would it make sense to have a PL with dynamic scope and static typing? (2 points)
8. 
   a. How can a programming language support programmers so that programs do not have run-time errors due to dangling references. (3 points)

   b. Many programming language allow programmer to redefine entities in nested blocks. Provide pluses and minuses to this feature. (4 points)

9. 
   a. For what kind of applications would you use a dynamic type language. Explain your answer. (4 points)

   b. For what kind of applications would you use a statically type language. Explain your answer. (4 points)

10. 
    a. State the differences between the following types: array, structs (or records), and sum of types. (8 points)

    b. In Java, how are arrays, records and sum-of-types provided? (6 points)
11. a. Explain how method execution is managed by the run-time system; and explain how the run-time system enforces the scope discipline used by the language. (6 points)

12. a. Name and discuss the purpose of the 3 memory areas available for programs during execution. (6 points)

b. If a method in a programming language returns a function as a value, where should that function be created so that the user can invoke after creation? Explain your answer. (4 points)
13.
   a. Define first-class values in a programming language. (6 points)

   b. Give two types in Java, one for primitives, one for composites, whose values are first-class. (2 points)

   c. Functions are not first class values in Java; explain why, and proceed to illustrate a way that we can simulate functions as first class values in Java. (10 points)
14. Briefly answer the questions below based on your chosen language of study: __________________________. One point each.
   a. language pragmatics

   b. variable semantics:

   c. scope discipline:

   d. typing disciple:

   e. kind of translators available:

   f. storage methods provided:

   g. language constructs which give rise to aliases:

   h. can programmer control memory management?

   i. garbage collector provided?

   j. parameter passing method(s):

   k. enumeration types provided?

   l. composite built-in types for homogeneous data:

   m. module mechanisms

   n. type safety

   o. binding time for bodies of methods

   p. latest binding time for binding of an array’s size.
q. type equivalence methods used by language (if more than one state which methods are used for which types)

r. automatic type conversions available

s. explicit casting provided

t. first class values

u. second class values

v. If language supports some form of polymorphism explain mechanism

w. expression evaluation (eager/lazy)

x. expressions that have to be evaluated by some form of lazy evaluation

y. order of evaluation of expressions

z. order of evaluation of arguments to methods

aa. generic programming for methods

ab. generic programming for data structures