



Programming I (Python)

Assignment 1

1. Which of the following are true about Lisp programming language:
 1. Strictly typed
 2. Statically typed
 3. Dynamically typed
 4. Safe
2. What are the features of a first class object in a programming language?
 1. Can be called as a procedure
 2. Can be passed as a parameter to a function
 3. Can be used as a type
 4. Can be returned from a function as a value
 5. Can be stored in a data-structure
 6. Can be imported as a module
3. Which of the following are true about OCaml commands:
 1. They are pure commands.
 2. They always produce a value.
 3. They may produce a value.
 4. They do not produce a value.
4. Which of the below are true about OCaml type system?
 1. Statically typed
 2. Dynamically typed
 3. Implicitly typed
 4. Explicitly typed
 5. Both implicitly and explicitly typed
5. Which of the following statements is true about OCaml expressions:

1. OCaml expressions can't have side-effects.
 2. OCaml expressions evaluate to a single value.
 3. Every expression has exactly one type.
 4. The type of the expression depends of the values evaluated so far.
6. The name of the OCaml debugger is:
1. ogdb
 2. odebug
 3. ocamldebug
 4. ocamlgdb
7. Which of the following are valid variable names in OCaml:
1. abc
 2. ab_c
 3. Abc
 4. Ab_c
 5. ab-c
 6. ab1
8. When we use one of the OCaml compilers to compile an OCaml program `program.ml`, the compiler name and object code file name are related as follows:
1. `ocamlc` \mapsto `program.mlo`
 2. `ocamlopt` \mapsto `program.mlo`
 3. `ocamlc` \mapsto `program.mlc`
 4. `ocamlc` \mapsto `program.cmo`
 5. `ocamlc` \mapsto `program.cmx`
 6. `ocamlopt` \mapsto `program.cmo`
 7. `ocamlopt` \mapsto `program.cmx`
9. In OCaml, When an expression is evaluated, which of following things may happen:
1. It may evaluate to a value of the same type as the expression.
 2. If typechecked successfully, it will never raise an exception.
 3. It may not terminate.
 4. It is guaranteed to terminate.
10. Which of the following are true about OCaml programs:
1. OCaml programs must always be written on the top loop.
 2. OCaml programs must be written in a file like in C.
 3. OCaml programs must always be compiled explicitly before being executed.

4. OCaml compiler always produces native machine code as output as in C.
5. OCaml compiler always produces bytecode as output as in Java.
6. OCaml compiler can be used to produce either machine code or byte code as per user preference.