**SECTION A**

1. Describe how to implement a compiler for a language in the same language (“bootstrap- ping”).
2. Invent a syntax for an APL-like matrix-based language that uses ordinary characters.
3. Write a list of interesting operations on strings and compare your list with the predeﬁned operations of SNOBOL and Icon.
4. Write a list of interesting operations on sets and compare your list with the predeﬁned operations of SETL.
5. Simulate a (universal) Turing machine in several programming languages.

**SECTION B**

1. Translate (part of) the BNF syntax of C or Ada into syntax diagrams.
2. Write a program in Pascal or C that compiles and executes, but computes the wrong answer because of a comment that was not closed.
3. Even if Ada used the style of comments used in C and Pascal, bugs caused by not closing comments would be less frequent. Why?
4. In most languages, keywords like begin and while are reserved and may not be used as identiﬁers. Other languages like FORTRAN and PL/I do not have reserved keywords. What are the advantages and disadvantages of reserved words?

**SECTION C**

1. Study your compiler’s documentation and list the optimizations that it performs. Write programs and check the resulting object code for the optimizations.
2. What information does the debugger need from the compiler and linker?
3. Run a proﬁler and study how it works.
4. How can you write your own simple testing tool? What inﬂuence does automated testing have on program design?
5. AdaS is an interpreter for a subset of Ada written in Pascal. It works by compiling the source into P-Code and then executing the P-Code. Study the AdaS program and write a description of the P-Code machine.

**SECTION D**

1. Read your compiler documentation and list the precisions used for the different integer types.
2. Write 200 + 55 = 255 and 100 −150 = −50 in two’s complement notation.
3. Let a take on all values in the ranges 50…,56 and −56..−50,and let b be either 7 or−7. What are the possible quotients q and remainders r when a is divided by b? Use both deﬁnitions of remainder (denoted rem and mod in Ada), and display the results in graphical form. Hint: if rem is used, r will have the sign of a; if mod is used, r will have the sign of b.

1. What happens if you execute the following C program on a computer which stores short int values in 8 bits and int values in 16 bits?

 Cshort int i; int j = 280; for (i = 0; i ¡ j; i++) printf(”Hello world”);

1. If a non-standard representation of an enumeration type is used, how would you implement the Ada attribute T’Succ(V)?
2. What will the following program print? Why?

 Cint i = 2; int j = 5; if (i & j) printf(”Hello world”); if (i && j) printf(”Goodbye world”);

1. What is the value of *i* after executing the following statements?

 Cint i = 0; int a[2] = {10,11}; i = a[i++];

1. C and C++ do not have an exponentiation operator; why?
2. Show how modular types in Ada 95 and unsigned integer types in C can be used to represent sets. How portable is your solution? Compare with the set type in Pascal.
3. Given an arithmetic expression such as:

 (a + b) \* (c + d)

 Java speciﬁes that it be evaluated from left to right, while C++ and Ada allow the compiler to evaluate the sub-expression in

 any order. Why is Java stricter in its speciﬁcation?

1. Compare the ﬁnal construct in Java with constants in C++ and Ada.

**SECTION E**

1. Does your compiler pack record ﬁelds or align them on word boundaries?
2. Does your computer have a block-copy instruction, and does your compiler use it for array and record assignments?
3. Pascal contains the construct with which opens the scope of a record so that the ﬁeld names can be used directly:

 Pascaltype Rec = record Field1: Integer; Field2: Integer; end; R: Rec;

 with R do Field1 := Field2;(\* OK, direct visibility \*)

 What are the advantages and disadvantages of the construct? Study the Ada renames con- struct and show how some of

 the same functionality can be obtained. Compare the two constructs.

1. Explain the error message that you get if you try to assign one array to another in C:

 Cint a1[10], a2[10]; a1 = a2;

1. Write sort procedures in Ada and C, and compare them. Make sure that you use attributes in the Ada procedure so that the procedure will work on arrays with arbitrary indices.
2. Which optimizations does your compiler do on array indexing operations?
3. Icon has associative arrays called tables, where a string can be used as an array index:

 count[”begin”] = 8; Implement associative arrays in Ada or C.

1. Are the following two types the same?

 Adatype Array Type 1 is array(1..100) of Float; type Array Type 2 is array(1..100) of Float;

 Ada and C++ use name equivalence: every type declaration declares a new type, so two types are declared. Under structural

 equivalence (used in Algol 68), type declarations that look alike deﬁne the same type. What are the advantages and

 disadvantages of the two approaches?

1. An array object of anonymous type (without a named type) can be deﬁned in Ada. In the following example, is the assignment legal? Why?

 AdaA1, A2: array(1..10) of Integer; A1 := A2;

1. Compare the string processing capabilities of Ada 95, C++ and Java.

**SECTION F**

1. Does your compiler implement all case-/switch-statements the same way, or does it try to choose an optimal implementation for each statement?
2. Simulate a Pascal repeat-statement in Ada and C.
3. The original deﬁnition of FORTRAN speciﬁed that a loop is executed at least one time even if the value of low is greater than the value of high! What could motivate this design?
4. The sequential search in C:

 Cwhile (s[i].data != key) i++;

 might be written as follows:

 Cwhile (s[i++].data != key) ; /\* Null statement \*/

 What is the difference between the two computations?

1. Suppose that Ada did allow an index variable to exist after the scope of the loop. Show how optimization of a loop would be affected.
2. Compare the code generated for a search implemented using a break- or exit-statement with the code generated for a sentinel search.
3. Write a sentinel search using do-while rather than while. Is it more efﬁcient?
4. Why did we put the sentinel at the beginning of the array rather than at the end?
5. (Scholten) The game of Go is played with stones of two colors, black and white. Suppose that you have a can with an unknown mixture of stones and that you execute the following algorithm:

 Adawhile Stones Left in Can loop Remove Two Stones(S1, S2); if Color(S1) = Color(S2) then Add

 Black Stone; else Add White Stone; end if; end loop;

 Show that the loop terminates by identifying a value which is always decreasing but always non-negative. Can you say

 anything about the color of the last stone to be removed? (Hint: write a loop invariant on the number of white stones.)