1. Which language is considered the first high-level programming language? (mark one, 4 points)
   (a) Pascal
   (b) C++
   (c) Fortran
   (d) Lisp

2. Which phases are part of the front-end of a compiler? (mark one or more, 4 points)
   (a) Code improvement
   (b) Semantic analysis
   (c) Syntax analysis
   (d) Target code generation

3. What is the value of the Scheme expression (cons 0 (member 3 '(1 2 3)))? (mark one, 4 points)
   (a) (0 1 2 3)
   (b) (0)
   (c) 3
   (d) (0 3)

4. What is a special form in Scheme? (mark one, 4 points)
   (a) A function with a special name
   (b) A function with special evaluation rules for arguments
   (c) A list
   (d) None of the above

5. A recursive descent parser is... (mark one or more, 4 points)
   (a) A top-down parser
   (b) A bottom-up parser
   (c) A parser for LR grammars
   (d) A predictive parser for LL grammars

6. The gdb tool is a... (mark one, 4 points)
   (a) A compiler
   (b) An interpreter
   (c) A debugger
   (d) A linker
7. What is a fixed format language? (mark one, 4 points)
   (a) A language with a fixed number of keywords
   (b) A language in which the relative positions of tokens with respect to each other is important rather than the position of the tokens on the page
   (c) A language in which indentation is significant, hence the amount of spacing is meaningful and influences the execution of a program
   (d) A language with formatted read and write constructs

8. Consider the program below:

   ```
   procedure P(A:real)
   var X:real;
   procedure Q(B:real)
   var Y:real;
   begin (* body of Q *)
   ...
   end (* body of Q *)
   procedure R(C:real)
   var Z:real;
   begin (* body of R *)
   ... <= (*)
   end (* body of R *)
   begin (* body of P *)
   ...
   end (* body of P *)
   ```

   What is the complete reference environment at the (*) location in this program? (mark one, 4 points)
   (a) A, C, P, Q, R, X, Z
   (b) A, C, P, R, Z
   (c) A, B, C, P, Q, R, X, Y, Z
   (d) C, R, Z

9. Fill in the blanks. Choose from: language design, language implementation, programing writing, compile, link, load, and run. (8 points)
   (a) Dynamic semantic errors are detected at _________________ time.
   (b) Static semantic errors are detected at _________________ time.
   (c) The syntax of a new programming language is determined at _________________ time.
   (d) Multiple object codes are merged into one executable at _________________ time.

10. What is a first-class object in a programming language? (7 points)
11. Draw the typical *subroutine frame* layout and explain the functionality of the frame slots. (10 points)

12. Explain *heap allocation*. Describe two common block allocation selection algorithms, two algorithms that maintain pools of free blocks, and two fragmentation problems. (12 points)
13. Credit card numbers are validated using the LUHN formula. The formula combines the digits of a credit card number in a specific way to compute a total sum of which the mod 10 value must be zero for valid card numbers. Consider the following (incomplete) augmented grammar to verify credit card numbers with a simplified LUHN algorithm:

\[
\begin{align*}
\langle \text{cardno} \rangle & \rightarrow \langle \text{digits} \rangle \\
\text{cardno.ok} & := (\text{digits}.\text{len} == 16 \&\& \text{digits}.\text{sum} \mod 10 == 0) \\
\langle \text{digits}_1 \rangle & \rightarrow \langle \text{digit} \rangle \langle \text{digits}_2 \rangle \ldots \\
\langle \text{digits} \rangle & \rightarrow \langle \text{digit} \rangle \ldots \\
\langle \text{digit} \rangle & \rightarrow 0 \quad \text{digit}.\text{val} := 0 \\
\langle \text{digit} \rangle & \rightarrow 1 \quad \text{digit}.\text{val} := 1 \\
\langle \text{digit} \rangle & \rightarrow 2 \quad \text{digit}.\text{val} := 2 \\
\langle \text{digit} \rangle & \rightarrow 3 \quad \text{digit}.\text{val} := 3 \\
\langle \text{digit} \rangle & \rightarrow 4 \quad \text{digit}.\text{val} := 4 \\
\langle \text{digit} \rangle & \rightarrow 5 \quad \text{digit}.\text{val} := 5 \\
\langle \text{digit} \rangle & \rightarrow 6 \quad \text{digit}.\text{val} := 6 \\
\langle \text{digit} \rangle & \rightarrow 7 \quad \text{digit}.\text{val} := 7 \\
\langle \text{digit} \rangle & \rightarrow 8 \quad \text{digit}.\text{val} := 8 \\
\langle \text{digit} \rangle & \rightarrow 9 \quad \text{digit}.\text{val} := 9
\end{align*}
\]

(a) The number of digits must be 16 and the sum of the digits mod 10 must be zero. Complete the semantic rules above to compute the \text{digits}.\text{len} and \text{digits}.\text{sum} values for validation. (8 points)

(b) Which attributes are synthesized or inherited? Note: \text{cardno.ok} is synthesized. (3 points)

<table>
<thead>
<tr>
<th></th>
<th>synthesized</th>
<th>inherited</th>
</tr>
</thead>
<tbody>
<tr>
<td>\text{cardno.ok}</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>\text{digits.len}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>\text{digits.sum}</td>
<td></td>
<td></td>
</tr>
<tr>
<td>\text{digit.val}</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

14. Consider the following program:

\[
\begin{align*}
x & : \text{integer} /* \text{global} */ \\
\text{procedure one} & \\
x & := 1 \\
\text{procedure two} & \\
x & : \text{integer} \\
x & := 2 \\
\text{one()} & \\
\text{write_integer(x)} & \\
\text{begin} /* \text{main program} */ \\
\text{two()} & \\
\text{end} /* \text{main program} */
\end{align*}
\]

(a) Assuming that the programming language uses static scoping, what value does this program print? (5 points)

(b) Assuming that the programming language uses dynamic scoping, what value does this program print? (5 points)
15. Consider the following Scheme function:

```
(define f
  (lambda (lst)
    (cond
      ((null? lst)) 0)
      ((number? (car lst)) (+ 1 (f (cdr lst))))
      (else (f (cdr lst))
      )
    )
  )
```

Explain what the function `f` computes for lists. Note: consider `(f '(1 a b 2))` for example. (10 points)