1. True or False. A semicolon is mandatory in the subscripting of V1:
   Ans. __________

   STRUCTURE Q:
   1   A,
   2   V1 ARRAY(2,5) MATRIX,
   2   V2 ARRAY(2,5) VECTOR;
   DECLARE Q Q-STRUCTURE(30) INITIAL(10#(9#1),10#(3#2));
   V1$(10;) = V1$(20;);

2. Based on the preceding structure, how many scalars result from the subscript form:
   Q.V1$(3 AT 10; 1,* : *,2);
   Ans. __________

3. True or False. Based on the preceding structure, the colon is optional in the
   subscripting of V1: V1$(*; 2,4:)
   Ans. __________

4. Based on the preceding structure, what data type and organization is:
   V1$(1 TO 10; *,3 AT 2: *,2)
   Ans. __________

5. True or False. In order to assign or compare two major or minor structures, tree-
   equivalency is mandatory.
   Ans. __________

6. True or False. The trailing semicolons are optional in the assignment in the
   following case:
   Ans. __________

   STRUCTURE G:
   1   I  INTEGER,
   1   S;
   DECLARE G G-STRUCTURE(100);
   G$(3;),G$(9;),G$(10;) = G$(51;);

7. True or False. In the structure template in question 6, S is a minor node.
   Ans. __________

8. True or False. Two structures can be compared for greater-than (> or less-than (<)
   if and only if they are: 1) tree equivalent, and 2) all terminals are unarrayed integers or
   scalars.
   Ans. __________
9. True or False. Two structures (or minor structures) can be compared in \textsc{while} or \textsc{until} clauses added on to iterative or discrete \textsc{do for} loops. Ans. \underline{__________}

10. True or False. Two structures (or minor structures) are equal if and only if all corresponding terminals are equal – and, of course, they must be tree-equivalent. Ans. \underline{__________}

11. True or False. If a major or minor structure is passed as an input or assign argument to a \textsc{procedure}, then each of the structures must be tree-equivalent to the corresponding formal parameter. Ans. \underline{__________}

12. True or False. If a multicopied major structure, multicopied minor structure, or multicopied structure terminal is passed as an input argument to a \textsc{procedure}, then the argument must either be left with “full copyness” (i.e., no copy subscripting at all), or subscripting must be supplied that specifies only a single copy. Ans. \underline{__________}

13. True or False. In the following example \textsc{var1} is a properly formed input to \textsc{proc1}. Ans. \underline{__________}

\begin{verbatim}
STRUCTURE ZZ:
  1 A,
    2 \textsc{var1} MATRIX(5,5),
        2 \textsc{cvar} CHARACTER(20);
DECLARE W ZZ-STRUCTURE(10) INITIAL(1,*);
CALL PROC1(W.A.VAR1$(3 AT 5;));
\end{verbatim}

14. True or False. In the preceding example the subscripted \texttt{W.A.VAR1} form would be legal in an \texttt{assign} list. Ans. \underline{__________}

15. True or False. A multicopy structure acts like a 1-dimensional array – for example, an input parameter can be defined to be a structure with * copies which would allow the actual number of copies passed to be determined at run time by using the \texttt{size} function. Ans. \underline{__________}

16. True or False. A \textsc{hal/s} \texttt{function} can be defined to be of \texttt{structure} type. It is essential, however, that the structure template be defined “prior” to the definition of the function (e.g., in a \texttt{compool} or at \texttt{program} level). The following \texttt{function} header is correct: Ans. \underline{__________}

\begin{verbatim}
STRFUNC: FUNCTION(I1,I2) Q-STRUCTURE(10);
\end{verbatim}
17. True or False. Although a STRUCTURE function has a template and acts like a real STRUCTURE in the right-hand-sides of assignments or compares, unlike a real structure no reference may be made to any of its minor structures or terminals.
   Ans. __________

18. True or False. Arrays can be added to one another (or compared for equality or non-equality) as long as “arraynesses” match. Such arrayness is defined as an n-tuple wherein the first number represents the number of dimensions and the remaining integers define the legal range of each dimension, e.g., (3:2,10,4). Moreover, the first dimension of “arrayness” can arise as a result of the data item residing in a multicopy structure.
   Ans. __________

19. True or False. It is not necessary for all variables in an arrayed expression to have “arrayness”.
   Ans. __________

20. True or False. HAL/S allows one array to be set equal to the product of two other arrays provided all arraynesses match. Also, the following example will compile:
   Ans. __________

   DECLARE ARR1 ARRAY(10,4,6) SCALAR INITIAL(0);
   DECLARE ARR2 ARRAY(10,10,10) INTEGER DOUBLE INITIAL(0);
   DECLARE ARR3 ARRAY(12,8,11) SCALAR DOUBLE INITIAL(0);
   ARR3$(1 TO 10,4 AT 3,6 AT 2) = ARR1 ARR2$(*,4 AT 1,2 TO 7);

21. True or False. Arrayness is a function of subscripting as is the choice of an annotation mark placed over the variable in the output listing.
   Ans. __________

22. True or False. Any arrayness remaining after the subscripting operation is termed ‘residual arrayness’. If the subscripting results in a 1-dimensional array with only 1 element, however, we say that the NO arrayness is present.
   Ans. __________

23. True or False. The occurrences of arrays in the output writer listing are enclosed in curly braces ({}).
   Ans. __________

24. What is the annotation symbol placed over a STRUCTURE?
   Ans. __________

25. True or False. To build a COMSUB (separately compiled function) FUNCTION of STRUCTURE type, the structure template would necessarily have to be defined in an INCLUDE.
   Ans. __________