Logics Test

This test is designed to measure general technical and logistical skills. For this reason, it employs a language with which you are unlikely to be familiar. All information about the language necessary to complete the test is contained within the test itself.

Each page of the test contains a description of a language feature or characteristic, along with a question that tests your understanding of that feature or characteristic. Subsequent questions build upon features and descriptions in earlier parts of the test, which sometimes need to be applied in a complex way. There are no trick questions, although there are subtle applications of the information contained in the description portions. There is only one correct answer to each question.

This test takes approximately 30-45 minutes to complete. Please write your answers on the answer sheet provided. Work out your solutions on the blank sheets attached to the end of this test.

PLEASE DO NOT WRITE ON THIS TEST.

QUESTION #1: Numeric Constants

Numeric constants are integers which range in magnitude from 0 to 999999999999999 and may be signed or unsigned. Examples of numeric constants are:

0   -13   1300   1   14567

Some invalid numeric constants are:

1.2   not an integer
$1   not numeric
1000000000000000   too large

Question: Which of the following is not a legal numeric constant?

A.   10.2
B.   -0
C.   00000007
D.   89

QUESTION #2: String Constants

A string constant is a sequence of 0 to 255 characters chosen from the standard ASCII 64 character set.

Any legal character (alpha or numeric) may be used as a string and is designated as such when included in quotation marks (i.e. "ABC").

A string constant can also consist of zero characters (that is, quotation marks around no characters), which is referred to as the "nill" or "null" string. Some examples are:

"ABCDEF"   "123456"   "!!!$$"   ""

When string constants are read in or written out, the quotation marks are omitted.

Question: What kind of a constant is this: "4"

     A.   Numeric
     B.   String
     C.   I can't tell - it could be either string or numeric

QUESTION #3: Variable Names

A constant retains its value, by definition. The value of a variable may be changed in the course of a program. A variable may take on any value that is allowable for either a string or numeric constant.

Unlike some other programming languages (e.g. FORTRAN), the type of variable is not determined by the name or any kind of declaration. Variable names are in the format of 1 alphabetic character, followed by any number of other alphanumeric characters. Here are some examples of acceptable variable names:

     A   ABC   A12   A0

If a variable is set equal to a string value, such as "##$", it will be a string variable. The very same variable may be converted to a numeric variable by setting it to a numeric constant.

Question: Which of the following variable names can have a string value?

    A.   STG
    B.   S3
    C.   I
    D.   A and B (above)
    E.   A and C (above)
    F.   B and C (above)
    G.   A, B and C (above)
    H.   None of the above.

QUESTION #4: Arrays

In addition to scalar variables, there are also single dimension arrays, among others. The rules for naming arrays are the same as the rules for naming variables.

Unlike almost all other programming languages, there is no need (and no capability) for declaring array dimension. An element of an array is undefined until you put something into it. The only space used in the system is for those array elements which are defined. There is also no restriction on the data types of different array elements. Thus, you can have string and numeric values within the same array.

Suppose the only variables defined are:

A   = 0
A(0)   = 1
A(1)   = 32767
A(100)   = "NINETY-NINE"
A(32767)   = "THIS IS A STRING"

Question: What is the value of A(A(A+1))?

        A.   0
        B.   32767
        C.   1
        D.   "THIS IS A STRING"
        E.   Undefined

QUESTION #5: Numeric Arithmetic Operators

Operators connect constants and variables to form expressions. The arithmetic operators include:

+   Addition   1+2=3
-       Subtraction   3-2=1
*   Multiplication 2*3=6
/   Division   3/2=1 (Integer division only)
&   Minimum   2&3=2
!   Maximum   2!3=3
Modulo   13#5=3 (Remainder after division)

An important feature of these, and all operators, is that evaluation in an expression takes place STRICTLY FROM LEFT TO RIGHT. Parentheses may be used to group parts as in algebra.

The & (minimum), ! (maximum), and # (modulo) operators are probably less familiar than the other four operators. Below are some examples of how they can be used:

X#10 =   Remainder of X/10
X&10 =   X if X is less than 10, otherwise 10
X!10 =   X if X is greater than 10, otherwise 10
X!1&10   =   X if X is between 1 and 10, 1 is X is less than 1,
                      and 10 if X is greater than 10
-X!X =   Positive value of X

Question: Now for an example. What is 1+1*3/2+7#12?

        A.   9
        B.   7
        C.   0
        D.   10
        E.   -131071
        F.   None of the above

QUESTION #6: Numeric Comparison Operators

There are three operators for comparing numbers:

<   Less than
>   Greater than
=   Equals

You can combine arithmetic and comparison operators in an expression. True evaluates to the "rubout" character (binary, all ones), and False evaluates to "" (nill). Thus, 1+2=3 evaluates to "rubout" and 1+3<2 evaluates to "" (nill).

Question: Suppose A is a variable with a non-zero numeric value. Which of the following is false?

      A.   A + A=A*2
      B.   (A#A)>(A/A)
      C.   A*A>0
      D.   A#A=0

QUESTION #7: Boolean And

Simple in concept, though sometimes complicated in practice, are the Boolean operators & (AND), ! (OR), and ' (NOT). The symbol & is the Boolean operator for ANDing.

If applied against logical "true" and "false" statements, both statements must be "true" in order for the AND statement to be true. That is, if the variables A and B are both true, the statement A&B is true, while if either A or B (or both A and B) are false, the statement A&B is false.

Question: Assume the variable A is true, the variable B is true, and the variable C is false. Which of the following is false?

        A.   A&B&C
        B.   B&B
        C.   A&B
        D.   None of the above

QUESTION #8: Boolean Or

The ! is used as the Boolean OR operator.

The logical OR appears between two values or expressions. If either expression is true, the entire statement is evaluated as true. For example, in the expression A!B if either variable is true (or if both variables are true), the expression is evaluated as true.

Question: Assume that the variable A is true, the variable B is true, and the variable C is false. Which of the following is false?

  A. A!C
  B. A&C!B
  C. C!C!B
  D. All of the above
  E. None of the above

QUESTION #9: Boolean Not

The operator ' (apostrophe) is used to represent the Boolean NOT. If it is applied to an integer or variable it re-evaluates that integer or variable to a "nill" (note, not a zero).

Question: If you were asked for '17 you would answer:

      A.   True
      B.   " "
      C.   0
      D.   14
      E.   17
      F.   10001

QUESTION #10: Not Again

The NOT can also be used to modify comparison operators, e.g. 1'>2 (one not greater than two), 2'<1, 1'=2, and A#B'>B are true.

Question: Which of the following is false?

       A.   '(A=A)
       B.   A'<A
       C.   A'=('A)
       D.   'A'=A

QUESTION #11: Unary Operators

The - (minus) and the ' (not) are also referred to as "Unary" operators and may be used in arithmetic expressions. When used in this way, they apply to the value immediately to their right, BEFORE it is evaluated in the expression.

In arithmetic expressions, the "nill" string is evaluated as a zero, while in logical expressions it is evaluated as a false. For example, the expression '1 is evaluated as a nill; when it is used in the arithmetic expression 1+'1 the value of the expression would be 1, but when used logically in the expression 0='1, the expression is false.

Other examples:

-3=-3   '1=""   '10+0=0   2+3+'3*-3=-15

Question: One of the expressions below is false, which one?

    A.   4+'4='4+4
    B.   '('5)=5
    C.   1+'(-3)'>4
    D.   -5-5=2+'8*-5
    E.   1!2!4!8&'15+0=0

QUESTION #12: String Concatenation

Concatenation means linking together in a series or chain The period (.) is the concatenation operator.

For example:

"HI "."THERE" is the same as "HI THERE"
"HI"." THERE" is the same as "HI THERE"

Question: If GOD has the value "HOLY" and HOH has the value "WATER", how do you make "HOLY WATER"?

>
>   A.   GOD.WATER
>   B.   HOLY.WATER
>   C.   HOLY." ".WATER
>   D.   GODHOH
>   E.   .GOD.HOLY
>   F.   GOD." ". HOH
>   G.   GOD.HOH
>   H.   GOD.. HOH

QUESTION #13: Integer Concatenation

You can also concatenate integers. For example:

$5.5 = 55$ $678.910 = 678910$

Question: What is the value of 1.2+3.4?

A.   0
B.   4.6
C.   46
D.   154
E.   1234
F.   Not allowed - illegal syntax

QUESTION #14: Data Conversion

You can mix integers and non-integers in a single expression. For
example:

       123."A"="123A"   "4A"+123=127   "5.5"+5=10

Non-integers are converted to integers as required by the operators. The non-integer converts
to an integer starting with the leftmost character and continuing until encountering a non-
numeric character. In the second example above, "4A" converts to 4; in the third example,
"5.5" converts to 5.

Question: What is the value of "1.2"+3.4?

     A.   4.4
     B.   4.6
     C.   44
     D.   46.2
     E.   154
     F.   None of the above

QUESTION #15: More on Data Conversion

Question: Suppose A="JOHN", B="JANE", AND C="3DOES". What is the value of 0.A+B+C-3?

       A.   "0JOHNJANEDOES"
       B.   JOHNJANE3DOES
       C.   0
       D.   3
       E.   A and D
       F.   B and C
       G.   All of the above
       H.   None of the above

QUESTION #16: Set and Kill

Variables are defined dynamically. When you log-in to a system, no variables are defined. In the course of running, a program will define variables via the READ (R) and SET (S) commands, and undefine variables via the KILL (K) command. Notice that commands can be abbreviated to one letter.

As you know, there are two kinds of variables - string and numeric. A variable may be defined via a READ command, in which case it is automatically a string. The other way a variable can be defined is by the SET command which can define it as either string or numeric.

The syntax for the SET command is:

S arg1,arg2...argx

S VAR=expression   Assigns the value of the expression
                          to the variable VAR. The data type
                          of VAR is determined by the expression.

S VAR1=exp1,VAR2=exp2...   Assigns values to multiple variables.

S X=Y   Assigns the value of variable Y to
                          variable X.

Question: If I say: R Z S X=Z, Y=X
              What type of variables are X, Y and Z?

        A.   All string
        B.   All numeric
        C.   X, Y numeric; Z string
        D.   X numeric; Y, Z string
        E.   X numeric; Y, Z undefined
        F.   Can't tell without more information

QUESTION #17: More on Set

Question: What is the value of A after executing the following?

S A=1, B=1, C=1, A=1+A+(A-1)

A.   5
B.   3
C.   2
D.   1

QUESTION #18: More on Kill

There is a KILL command which removes a variable and its value from the symbol table, thus undefining it. The syntax for the KILL is:

K   A KILL command followed by 2 spaces kills all variables.
K VAR   The variable VAR is killed, if it exists.
K (VAR,VAS) All variables except VAR and VAS are killed.

The most common use of KILL is to free storage space. Another use is to flag conditions - if a particular value is defined, that means one thing; and if it is undefined, that means something else.

Question: K   S S="4",T=S+2,Z=T>6 K K, J, S
What variables are defined and what are their values?

    A.   T=6,Z=" "
    B.   T="6", Z=0
    C.   S="4",T=6,Z=0
    D.   S="4",T="6",X="0"
    E.   S="4",T="6",Z="FALSE"
    F.   S="4",T="S+2",Z="T>6"
    G.   No variables are defined

QUESTION #19: Final Set and Kill

Question: As a final quiz on SET and KILL, after executing the logic below, what variables are defined and what are their values?

S T=1,B="1",A=1,B=A=T=(T=B) K (A)

- A.   A=0
- B.   A=1
- C.   A=3
- D.   A=1,B="1",T=1
- E.   A=131071
- F.   B="1", T=1

QUESTION #20: $Piece

The $PIECE function is a useful string-valued function. It returns a sub-string of a base string as defined by a delimiter and one or more positional indicators. For example,

$P("ONE,TWO,THREE,FOUR,",2)="TWO"

This is because the base string (inside the quotes) is "ONE,TWO,THREE,FOUR,". The delimiter is the "," and the 2 is the positional indicator. This statement says, "take the second piece out of the variable with the piece defined as the portion of the variable placed between the defined delimiter (",")."

There is an optional argument that allows for the definition of the positional indicator to multiple places (i.e. "take the second through the fourth pieces" would be indicated by replacing the positional indicator with "2,4").

Further examples:

    S X= "JOHN DOE;1234 MAIN;ANYTOWN"
    $P(X;1)="JOHN DOE"
    $P(X;2)="1234 MAIN"
    $P(X;2,3)="1234 MAIN;ANYTOWN"

Question: Assume X is as shown above. What would appear if we executed the statement: WRITE $P(X;1,3)

    A.   JOHN
    B.   JOHN DOE
    C.   JOHN DOE; ANYTOWN
    D.   JOHN DOE1234 MAINANYTOWN
    E.   JOHN DOE; 1234 MAIN; ANYTOWN
    F.   An error
    G.   Nothing

Name:

Position:

Date/Time Test Returned:

PROGRAMMER TEST ANSWER SHEET

QUESTION	TOPIC	ANSWER
#1)	Numeric Constants	_____
#2)	String Constants	_____
#3)	Variable Names	_____
#4)	Arrays	_____
#5)	Numeric Arithmetic Operators	_____
#6)	Numeric Comparison Operators	_____
#7)	Boolean And	_____
#8)	Boolean Or	_____
#9)	Boolean Not	_____
#10)	Not Again	_____
#11)	Unary Operators	_____
#12)	String Concatenation	_____
#13)	Integer Concatenation	_____
#14)	Data Conversion	_____
#15)	More on Data Conversion	_____
#16)	Set and Kill	_____
#17)	More on Set	_____
#18)	More on Kill	_____
#19)	Final Set and Kill	_____
#20)	$Piece	_____