Basic I/O Concepts

In this chapter you learn how to write out data using formats that specify the exact way in which the numbers should be printed.

A format may be used to specify the exact manner in which a program will print out variables on the paper and also the number of significant digits to be printed. To understand the structure of FORMAT statement, we must know something about the output devices on which are data will be displayed. Common output devices include line printers, laser printers, and terminals. A line printer is a type of printer that takes its name from the fact that it prints output data a line at a time. Because it was the first computer output device, FORTRAN output specifications were designed with it in mind. Other more modern output devices are generally compatible with the older line printer so that the same output statement can be used for any device. The computer builds up a complete image of each line in memory before sending it to an output device. The computer memory containing the image of the line is called the output buffer. The output buffer for a line printer is usually 133 characters wide. The first character in the buffer is known as the control character; it specifies the vertical spacing for the line. The remaining 132 characters in the buffer contain the data to be printed on that line. The control character is not printed on the page. Instead, it provides vertical positioning control information to the printer.

FORTRAN control characters

	Action
1	Skip to new page
Blank	Single spacing
0	Double spacing
+	No spacing (print over previous line)

Never write a format descriptor that includes column 1 of the output line. Erratic paging behavior and incorrectly displayed values may result if you do so.

FORTRAN’s many format descriptors fall into four basic categories:

1. Format descriptors that describe the vertical position of a line of text.
2. Format descriptors that describe the horizontal position of data in a line.
3. Format descriptors that describe the output format of a particular value.
4. Format descriptors that control the repetition of portions of a FORMAT statement.

Symbols used with format descriptors

c Column number
d Number of digits to right of decimal place for real input or output.
m Minimum number of digits to be displayed.

n Number of spaces to skip
r Repeat count – the number of times to use a descriptor or group of descriptors.
w Field width – the number of characters to use for the input or output.

Integer Output – The I Descriptor

It is used to describe the display format of integer data. General form is

rlw or rlw.m

Integer values are right justified in their fields. (These values are printed out so that the last digit of the integer occupies the right-most column of the field.) If the integer is too large to fit into the field in which it is to be printed, then the field is filled with asterisks.

Real Output – The F Descriptor

It is used to describe the display format of real data. General form is

rFw.d

Real values are printed right justified within their fields. If necessary, the number will be rounded off before it is displayed. For example, pi = 3.141593 using the F7.3 format descriptor is displayed as bb3.142 If the variable pi is displayed with an F10.8 format descriptor, the resulting value will be 3.14159300. If the real number is too large to fit into the field in which it is to be printed, then the field is filled with asterisks.

Real Output – The E Descriptor

Real data can also be printed in exponential notation using the E descriptor. Scientific notation is a popular way for scientists and engineers to display very large or very small numbers. It consists of expressing a number as a normalized value between 1 and 10 multiplied by 10 raised to a power. General form is

rEw.d

In general the width of an E format descriptor field must satisfy the expression

$$w \geq d + 7$$

or the field may be filled with asterisks.

Logical Output – The L Descriptor

General form is rLw

The value of logical variable can be .TRUE. or .FALSE. The output of logical variable is either a T or an F, right justified in the output field.

Character Output – The A Descriptor

General form is rA or rAw

The rA descriptor displays character data in a field whose width is the same as the number of characters displayed.

The rAw descriptor displays character data in a field of fixed width w. If the width w of the field is longer than the length of the character variable, the variable is printed out right justified in the field. If the width of the field is shorter than the length of the character variable, only the first w characters of the variable will be printed out in the field.

Horizontal Positioning – The X and T Descriptor

The X descriptor has the form nX where n is the number of blanks to insert. It is used to add one or more blanks between two values on the output line.

The T descriptor has the form Tc where c is the column number to go to. It is used to jump directly to a specific column in the output buffer.

When using the T descriptor, be careful to make certain that your fields do not overlap.

We can also repeat whole groups of format descriptors by enclosing the whole group within parenthesis and placing a repetition count in front of the parenthesis. For example,

```
320  FORMAT ( 1X, I6,  2(I6, 2F10.2)  )
```

The Slash (/) descriptor

It sends current output buffer to the printer and starts a new output buffer. With slash descriptors, a single write statement can display output values on more than one line. Several slashes can be used together to skip several lines. It need not be separated from other descriptors by commas. However, you may use commas if you wish.