## Fortran carriage control alternatives

## Fortran carriage control alternatives

The Unix implementation of Fortran-77 does not handle the ANSI-standard "carriage control" codes. These codes are:

space: single space before printing0: double space before printing

1: eject page before printing

+: overprint the previous line

By default, Unix Fortran-77 automatically adds a newline after the last record of a formatted sequential write. In addition, all carriage control characters are output in column 1 instead of being interpreted by your terminal.

You can direct the output of your fortran program through an output filter, 'fpr', which interprets the carriage control characters output by your program. Use this command:

```
a.out | fpr
```

You can direct output to a file like this:

```
a.out | fpr > filename
```

If you want, you can then send the 'filename' file to a printer to get a printed copy of your output.

Try compiling the sample program below. Then use the commands above to demonstrate how the 'fpr' filter implements carriage control.

The Unix implementation of Fortran-77 also allows you to use the non-standard format specifier '\$' to suppress the newline at the end of the last record of a formatted sequential write operation. This is useful when implementing a prompting interface. This is an example of a prompting interface:

```
write(6,10)
10    format(1x,"Please enter data here: ",$)
    read(5,*) variable
```

## Sample program:

```
This is a sample program for 'f77' carriage control.
C
С
      Output should be run through the 'fpr' filter.
      write(6,10)
10
      format(1x, "[space] means single space before printing")
      write(6,20)
20
      format("0","0 means double space before printing")
      write(6,30)
30
      format("1","1 means eject page before printing")
40
      format(1x,"+ means overprint the previous line")
      write(6,50)
50
      format("+","xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx")
      st.op
      end
```