

Файловый указатель - аналог экранного курсора, текущая позиция в открытом файле. При открытии файла файловый указатель всегда находится в первой позиции первой строки, дальше он перемещается в зависимости от требований операторов READ или WRITE. Каждый новый оператор READ или WRITE начинает следующую запись (строка). Управление файловым указателем возможно также с применением форматного ввода-вывода.

Оператор BACKSPACE перемещает файловый указатель на одну строчку вверх. Для установления файлового указателя в первую позицию первой строки используется оператор REWIND.

REWIND (UNIT = N), REWIND N

BACKSPACE (UNIT = N), BACKSPACE N

Параметр UNIT задает номер устройства, с которым он идентифицирован в программе (имеет тип INTEGER). Служебное слово UNIT является необязательным.

Пример. Записать в файл с номером устройства 18 значения переменных A, B, C, и считать эти значения в переменные X, Y, Z.

```
WRITE (18 *) A, B, C
```

```
REWIND 18
```

```
READ (18 *) X, Y, Z
```

Оператор INQUIRE позволяет программе определить характеристики устройства (файла). Этот оператор имеет много возможностей и опций. В качестве примера, покажем только определение имя файла с номером устройства.

Пример. Определить, является открытым устройство 16? Если да, то определить имя файла и передать его переменной FNAME типа CHARACTER.

```
INQUIRE (UNIT = 16 OPENED = OPN)
```

```
IF (OPN) THEN! Логическая переменная OPN = .TRUE.
```

```
INQUIRE (UNIT = 16, FILE = FNAME)
```

```
WRITE (*, *) '***** Opened File ', FNAME
```

```
ELSE! Логическая переменная OPN = .FALSE.
```

```
WRITE (*, *) '***** Unit 16 do not open "
```

```
ENDIF
```

Оператор ENDFILE вносит запись «конец файла» в текущее файлового указателя.

```
ENDFILE N
```

N - номер устройства.

После установки записи «конец файла» оператор ENDFILE устанавливает указатель файла после записи конца файла. Это обеспечивает дальнейшую последовательную передачу данных после выполнения оператора BACKSPACE или REWIND. ENDFILE может скрывать все записи по вновь сделанным записью «конец файла».

Логическая функция EOF (N) возвращает .TRUE., Если файловый указатель находится на записи «конец файла» устройства с номером N и .FALSE., Если нет.

Пример. Определить, сколько записей в файле INPUT.TXT.

```
OPEN (1, FILE = 'INPUT.TXT ', STATUS = ' OLD ')
I = 0
DO WHILE (.NOT.EOF (1))
READ (1, *)
I = I + 1
END DO
WRITE (*, *) '*** Number of the records in INPUT.TXT:', I
```

Обмен данными между программными единицами возможен с помощью файлового интерфейса. Однако, следует помнить, что точность отображения данных в файле (на экране) ограничена по сравнению с точностью их виртуального сохранения при обработке. Хотя это и зависит от аппаратно-программного обеспечения ПЭВМ, но практически, следует рассчитывать на 7 значащих цифр для данных обычной точности.

Пример. В приведенной выше программе расчета перемещения U (м) диска, который вращается, реализовать передачу данных в функции FN с помощью файлового интерфейса.

```
PROGRAM WH2
INTEGER H
REAL MU
DATA R1 / .05 /, R2 / .1`3 /, E /2.1E11/, G / 9.8 /, GAM / 7.7E4 /, H / 2 /, MU / .3 /
DATA W1, W2, DW / 50., 150., 10. /
WRITE (*, *) '***** INPUT DATA ***** '
WRITE (*, *) 'R1 =', R1, "R2 = ", R2
WRITE (*, *) 'E =', E, 'G =', G
WRITE (*, *) 'GAM =', GAM, "H = ", H, " MU =", MU
WRITE (*, *) '***** SOLVED NEXT ***** '
WRITE (*, *) 'W (rad / c) U (m)'
OPEN (1, FILE = 1.TXT, STATUS = 'REPLACE')
WRITE (1, *) MU, GAM, H, G, E
OPEN (2, FILE = 2.TXT, STATUS = 'REPLACE')
DO W = W1, W2, DW
REWIND 2
WRITE (2, *) W

$$U = FN () * R2 / 8 * ((3 + MU) / (1 - MU) * (R1 ** H + R2 ** H) +$$


$$+ (3 + MU) / (1 - MU) * (R1 ** H - R2 ** H))$$

WRITE (*, *) W, U
ENDDO
CLOSE (1)
CLOSE (2)
STOP
CONTAINS
FUNCTION FN ()
REWIND 1
```

```
REWIND 2  
READ (1, *) X1, X2, X4, X5, X6  
READ (2, *) X3  
FN = (1-X1 ** X4) * X2 * X3 ** X4 / X6 / X6  
END FUNCTION FN  
END
```