

Лабораторная работа 7

7.1 [#20]

Разработайте программу Split, которая символы из INPUT, предшествующие #, размещает в двух файлах: в файле Odds помещаются символы, занимающие нечетные позиции в INPUT, а в файле Evens помещаются символы, занимающие четные позиции в INPUT. Затем в OUTPUT выводятся сначала символы из Odds, а потом символы из Evens. Проведите сборку с тестированием каждого этапа, при необходимости добавив тестовые операторы.

Выполнение:

```
INPUT: 12345#  
OUTPUT: 13524
```

Проверьте на первом этапе изменение значения переменной-переключателя.

7.2 [#30]

Соберите программу SelectSort из приведенных ниже разделов проекта. Определите последовательность сборки, проведите тестирование каждого этапа сборки, при необходимости добавив тестовые операторы.

DP1.

```
PROGRAM SelectSort(INPUT, OUTPUT);  
{Сортирует символы, предшествующие #, из INPUT в OUTPUT.  
Программа выдаст ошибку, если в INPUT отсутствует #}  
VAR  
    Ch, Min: CHAR;  
    F1, F2: TEXT;  
BEGIN {SelectSort}  
    {Копировать INPUT в F1 и эхо в OUTPUT}  
    {Сортировать F1 в OUTPUT, используя стратегию SelectSort}  
END. {SelectSort}
```

DP1.1

```
BEGIN {Копировать INPUT в F1 и эхо в OUTPUT}  
    REWRITE(F1);  
    WRITE(OUTPUT, 'INPUT DATA:');  
    READ(INPUT, Ch);  
    WHILE Ch <> '#'  
    DO  
        BEGIN  
            WRITE(F1, Ch);  
            WRITE(OUTPUT, Ch);  
            READ(INPUT, Ch)  
        END;  
    WRITELN(OUTPUT);  
    WRITELN(F1, '#')  
END
```

DP1.2

```
BEGIN {Сортировать F1 в OUTPUT, используя стратегию SelectSort}
  WRITE(OUTPUT, 'SORTED DATA:');
  RESET(F1);
  READ(F1, Ch);
  WHILE Ch <> '#'
  DO { Ch <> '#' и Ch1 - первый символ F1}
    BEGIN
      {Выбираем минимальный из F1 b копируем остаток F1 в F2}
      WRITE(OUTPUT, Min);
      {Копируем F2 в F1}
      RESET(F1);
      READ(F1, Ch)
    END;
  WRITELN(OUTPUT);
END
```

DP1.2.1

```
BEGIN {Выбираем минимальный из F1 b копируем остаток F1 в F2}
  REWRITE(F2);
  Min := Ch;
  READ(F1, Ch);
  WHILE Ch <> '#'
  DO { Ch <> '#' и Ch1 - первый символ F1}
    BEGIN
      {Выбираем минимальный из (Ch, Min)
      копируем второй символ в F2}
      READ(F1, Ch)
    END;
  WRITELN(F2, '#');
END
```

DP1.2.2

```
BEGIN {Копируем F2 в F1}
  RESET(F2);
  REWRITE(F1);
  READ(F2, Ch);
  WHILE Ch <> '#'
  DO
    BEGIN
      WRITE(F1, Ch);
      READ(F2, Ch)
    END;
  WRITELN(F1, '#');
END
```

DP1.2.1.1

```
BEGIN {Выбираем минимальный из (Ch, Min)
      копируем второй символ в F2}
  IF Ch < Min
  THEN {Ch - минимальный из (Ch, Min)}
    BEGIN
```

```
        WRITE(F2, Min);  
        Min := Ch;  
    END  
ELSE {Min - минимальный из (Ch, Min)}  
    WRITE(F2, Ch);  
END
```