

Operator cycle implements the algorithm for DO loop with a precondition.

There are several types (Shooting) operator cycle DO. The most common use of the following: DO - label; DO - END DO; DO WHILE - END DO

The structure of the operator DO - next label.

```
DO M I = Istart, Iend, Istep
```

```
<Any number of operators loop body>
```

```
M CONTINUE
```

where DO - keyword (operator cycle); M - label last cycle operator body (such as an empty operator - operator CONTINUE); I - variable cycle (the cycle), which varies from the initial value to the final value Istart Iend increments Istep. Value Istep = 1 operates by default and may be absent.

Example. Counter as the cycle passes.

```
X = 0.
```

```
DX = .1
```

```
DO 20 I = 1, 200
```

```
Y = SIN (X) * COS (X)
```

```
WRITE (*, *) I, X, Y
```

```
X = X + DX
```

```
20 CONTINUE
```

DO operator is able to accurately track the termination condition of the loop only when the option cycle, its initial, final value and step changes are intact (with type INTEGER). Therefore, some legacy Fortran compilers for other types of options not provided. However, modern and more convenient Fortran previous example can be written as follows.

Example. The cycle is of type REAL.

```
DO 20 X = .0, 20., .1
```

```
Y = SIN (X) * COS (X)
```

```
WRITE (*, *) X, Y
```

```
20 CONTINUE
```

Keep in mind that for type REAL accuracy of calculations of the cycle, as well as any variable bit limited net. Such minor deviations from the "exact" can cyclically

sumuvatysya. Then compare the current (last) cycle I value and its final value Iend give questionable results.

In practice, this leads to the fact that the operator DO cycle may not always perform the last step of the cycle or make once. This is some compilers warn the user during compilation Vykhino code (Fig. 2.19).

Fig. 2.19. Warning compiler Silverfrost FTN95 for Microsoft® .NET and Win32 for use REAL type for the cycle DO

The latest operator the loop DO-label operator can not control DO, IF, GOTO, operators STOP, END, of course, can not be operators REAL, INTEGER, DATA. There may be, for example, operators WRITE, assignment operator, operator empty CONTINUE.

At one label may refer to several operators DO-loop tag.

Maybe make nested loops to one another, but constitute unacceptable cycles that overlap.

Example. Nested loops DO.

```
DO 10 T = 12.3, 706.8, .01
W = T + SIN (X)
DO 10 W = W0, WMAX, WSTEP
T = W + T
10 CONTINUE
```

Example. Cycles DO, crossed that you can not build.

```
DO 10 I = 1, 15
J = I + 25
DO 20 K = 3, 18
L = K + 25 + J
10 CONTINUE
WRITE (*, *) J, L
20 CONTINUE
```

Operator DO-label - is the first operator cycle that appeared in high-level programming languages. It requires a label, and therefore considered obsolete. However, he supported the use of any modern Fortran compiler.

The problem with labels operator decides to use cycle DO- END DO. The operator has

the following notation.

```
DO I = Istart, Iend, Istep
```

```
<Any number of operators loop body>
```

```
END DO
```

where DO - keyword (operator cycle); END DO - keyword (late cycle); I - variable cycle (the cycle), which varies from the initial value to the final value Istart Iend increments Istep. Value Istep = 1 operates by default and may be absent.

The source code example that was discussed earlier, this case will be as follows.

Example. Using the cycle operator DO-END DO.

```
DO X = .0, 20., .1
```

```
Y = SIN (X) * COS (X)
```

```
WRITE (*, *) X, Y
```

```
END DO
```

The cycle I, Istart its initial and final value Iend and Istep step in the DO-loop operators and label DO-END DO can be not only constants, but variables or as expressions.

Example. Variables and expressions in the operator DO.

```
DO X = W1, (A-C) / 2, GAMMA3
```

This cycle parameter X has an initial value in a variable W1, which should be razrahovana advance; expression (AC) / 2 - the final value of the cycle; Variable GAMMA3 - step change in the cycle.

If you use several operators DO-END DO, then DO keywords should match the number of END DO.