Assignment 1 (4 points)

In the lecture you saw flow charts describing for example the control flow of `for` with and without `break` or `continue` in the body:

```
init-statement
condition
statement
expression
false
for
```

```
init-statement
condition
statement
expression
break
break in for
```

```
init-statement
condition
statement
expression
continue
continue in for
```

Your task is to draw the corresponding flow charts for the `while` as well as the `do` loops. This means that you draw the following 6 flow charts: (i) `while`, (ii) `break in while`, (iii) `continue in while`, (iv) `do`, (v) `break in do`, (vi) `continue in do`

Assignment 2 (4 points)

Write a program `kdivisors.cpp` that inputs a natural number `k` (including 0) and outputs a list of all numbers `n` between 1 and 1000 with exactly `k` divisors (including 1 and `n`).

Judge Examples


<table>
<thead>
<tr>
<th>Number k =?</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>16 81 625</td>
<td></td>
</tr>
</tbody>
</table>
Assignment 3 – Skript-Aufgabe 50 (4 points)

Write a program dec2bin2.cpp that inputs a natural number \( n \) and outputs the binary digits of \( n \) in the correct order.

Hint: Find a way to “invert” the output of dec2bin.cpp.

Assignment 4 – Skript-Aufgabe 69 (4 points)

The number \( \pi \) can be defined through various infinite sums. Here are two of them.

\[
\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots \\
\frac{\pi}{2} = 1 + \frac{1}{3} + \frac{1}{3 \cdot 5} + \frac{1}{3 \cdot 5 \cdot 7} + \cdots
\]

Write a program for computing an approximation of \( \pi \), based on these formulas. Which formula is better for that purpose?
Note: Your program must have the prefix --- on each output line which does not contain the result. Furthermore, the result must be printed on a separate line. This makes sure that the Judge can correctly judge floating point numbers.

Judge Examples

--- Number of iterations =? 2
--- Pi is approximately 2.66667

--- Number of iterations =? 20
--- Pi is approximately 3.14159


Challenge - Skript-Aufgabe 56 (8 points)

(Submission by email.)

--- Explanation: Since this exercise involves floating point numbers we need to use a special code validator of the judge. This validator expects all lines where the result is not contained to start with --- as prefix; the validator ignores these lines. Thus the only line which shall not be ignored is the one with the result. And this remaining line is then compared with the reference solution up to a certain precision.