Lab 6.8.13.1 Disjoint compilation

Objectives

Familiarize the student with:

- Disjoint compilation
- Header files
- Printing on screen

Scenario

Write a program that prints two triangles: one is a normal triangle consisting of backslashes and the other is a Floyd's triangle. Remember to escape the backslash with a backslash (not a play on words!). A Floyd's triangle consisting of numbers in consecutive order: in the first row, we have only one number: 1; in the second row, two numbers: 2 3; in the third row: 4 5 6 and so on. Your program should ask the user for the size of both triangles (just one number - the triangles should be the same size). After that, your program should print both triangles. To print the Floyd's triangle, you may use the "%3d" format in *printf*. Divide your program into files: one file for the classic triangle function, one for the Floyd's triangle function, one header file with the prototypes of both functions, and finally a file with the *main* function. Practice adding and removing files from your program/project/solution. If you can, test it in different environments (different OS/different IDE/no IDE). Your version of the program must print the same result as the expected output.

Note: not all online compilers allow you to create a project of many files.

```c
#include <stdio.h>
/* your code that includes a header*/

int main()
{
 /* your code */
 return 0;
}
/* other files with your code */
```

Example input

```
15
```

Example output

```
\
\\
\\\
\\\\
\\\\\
\\\\\\
\\\\\\\
\\\\\\\\
\\\\\\\\\
\\\\\\\\\\
\\\\\\\\\\\
\\\\\\\\\\\\
\\\\\\\\\\\\\
```

```
   1
   2   3
   4   5   6
   7   8   9  10
  11  12  13  14  15
  16  17  18  19  20  21
  22  23  24  25  26  27  28
  29  30  31  32  33  34  35  36
  37  38  39  40  41  42  43  44  45
  46  47  48  49  50  51  52  53  54  55
  56  57  58  59  60  61  62  63  64  65  66
  67  68  69  70  71  72  73  74  75  76  77  78
  79  80  81  82  83  84  85  86  87  88  89  90  91
  92  93  94  95  96  97  98  99 100 101 102 103 104 105
```

Example input

```
5
```

Example output

```
5

\
\\
\\\
\\\\
```

```
   1
   2   3
   4   5   6
   7   8   9  10
```

