Problem 2: Expressions

Introduction

In this problem we consider arithmetic expressions that are composed of the following symbols:

- the four binary operators '`+`', '`-`', '`*`', and '`/`',
- the variables '`x`' and '`y`',
- the functions '`sin`' and '`cos`',
- opening and closing parenthesis '`(`' and '`)`'.

Here are some examples for valid expressions:

```
x+x*y+sin(x)    x+(x+x)      (((x)))      sin(x)+cos(y)
 (cos((x)))    ((x)*(x))   (x+y)*(x-y)   sin(sin(sin(x)))
```

Notice that in this problem '`-`' is a binary operator, unary use is not allowed. Therefore '`-x`' and '`x*-y`' are *not* valid expressions. The parentheses around the argument of '`sin`' and '`cos`' are obligatory, thus '`sin x`' or '`sin x*y`' are not valid expressions.

Your job is to write a program that, for a given length ℓ, generates *every* valid expression of length exactly ℓ. The output of your program is a text file containing every such expression exactly once. Each line of this text file should contain one expression, and the lines of the file should be sorted (in ASCII order). For example, the correct answer for $\ell = 3$ is

```
(x)
(y)
x*x
x*y
x+x
x+y
x-x
x-y
x/x
x/y
y*x
y*y
y+x
y+y
y-x
y-y
y/x
y/y
```

The answer for $\ell = 6$ is

```
cos(x)
cos(y)
sin(x)
sin(y)
```