A basic arithmetic expression is composed of characters from the set  $\{1, +, \times\}$  and parentheses. Almost every integer can be represented by more than one basic arithmetic expression. For example, all of the following basic arithmetic expression represent the integer 14:

Version: 1.0

$$1+1+1+1+1+1+1+1+1+1+1+1+1+1+1+1+1+1+1$$

$$((1+1)\times(1+1+1+1+1))+((1+1)\times(1+1))$$

$$(1+1)\times(1+1+1+1+1+1+1)$$

$$(1+1)\times(((1+1+1)\times(1+1))+1)$$

Describe and analyze an algorithm to compute, given an integer n as input, the minimum number of 1's in a basic arithmetic expression whose value is equal to n. The number of parentheses doesn't matter, just the number of 1's. For example, when n = 14, your algorithm should return 8, for the final expression above. The running time of your algorithm should be bounded by a small polynomial function of n.

## To think about later:

2 Suppose you are given a sequence of integers separated by + and - signs; for example:

$$1 + 3 - 2 - 5 + 1 - 6 + 7$$

You can change the value of this expression by adding parentheses in different places. For example:

$$1+3-2-5+1-6+7=-1$$
$$(1+3-(2-5))+(1-6)+7=9$$
$$(1+(3-2))-(5+1)-(6+7)=-17$$

Describe and analyze an algorithm to compute, given a list of integers separated by + and - signs, the maximum possible value the expression can take by adding parentheses. Parentheses must be used only to group additions and subtractions; in particular, do not use them to create implicit multiplication as in 1 + 3(-2)(-5) + 1 - 6 + 7 = 33.