

```

int: amount = 5034 ;
int: max_steps = 200 ;
var 1..max_steps: steps ;

array[1..max_steps] of var bool: multiply ; % if true multiply by
10 else add one.
array[1..max_steps] of var int: value ;

constraint value[1] = not multiply[1] ;
constraint value[steps] = amount ;
constraint forall(i in 2..max_steps) (value[i] = if multiply[i]
then value[i-1]*10 else value[i-1] + 1 endif) ;

solve minimize steps + sum([i|i in 1..max_steps where
multiply[i]]) ; % *10 gets there faster than +1.

output ["the number of operations required to achieve \ (amount) is
\ (steps).\n"] ++
["the operations are: "] ++ [if fix(multiply[i]) then "*10 " else
"+1 " endif|i in 1..fix(steps)] ;

% Running CalculatorWithTwoButtons.mzn
% the number of operations required to achieve 5034 is 15.
% the operations are: +1 +1 +1 +1 +1 *10 *10 +1 +1 +1 *10 +1 +1
+1 +1
% -----
% =====
% Finished in 511msec

```