Lab Assignment 2  week 4

EQUIPMENT: PC with ReTrO simulation system

EXPERIMENT 1
Build a full working CPU with 8bit memory (8bit opcodes, 8bit operands, 8bit addresses).

ALU: single accumulator, functions: nop / add / subtract

CU: All commands have 2 bytes, increment PC by 2 for every command (except branches).
- 0 x nop
- 1 v load constant acc := v
- 2 v add constant acc := acc + v
- 3 v subtract constant acc := acc - v
- 4 a store accumulator value to memory address a mem[a] := acc
- 5 a load memory value from address a in accumulator acc := mem[a]
- 6 a add memory value to accumulator acc := acc + mem[a]
- 7 a subtract memory value from accumulator acc := acc - mem[a]
- 8 a branch unconditionally to address pc+a pc := pc+a
- 9 v branch conditionally if acc = v to address pc+4 if acc=v then pc := pc+4 else pc := pc+2
- all other opcodes are "no operation"

EXPERIMENT 2
Write a program to calculate 1 + 2 + 3 ... + m, for a given value m.

\[ \text{result} = \sum_{i=1}^{m} i \]

Data locations: value \( m \) in memory location \$F0 \ (assume \( m \geq 1 \))
value \( \text{result} \) in memory location \$F1

Algorithm: clear result
loop: add \( m \) to \( \text{result} \)
decrement \( m \)
if \( (m=0) \) branch to loop

Note: Each of the commands in the pseudo-code sequence above requires several
assembly commands. E.g. "decrement \( m \)" requires the commands:
load mem[FO]; sub 1; store mem[FO];

Example:

<table>
<thead>
<tr>
<th>( m )</th>
<th>( \text{result} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>0</td>
<td>6</td>
</tr>
</tbody>
</table>