

13 The table shows assembly language instructions for a processor that has one register, the Accumulator (ACC).

Label	Instruction		Explanation
	Opcode	Operand	
	LDM	#n	Load the number n to the ACC
	LDD	<address>	Load the contents of the location at the given address to the ACC
	LDI	<address>	The address to be used is at the given address. Load the contents of this second address to the ACC
	ADD	<address>	Add the contents of the given address to the ACC
	ADD	#n	Add the number n to the ACC
	SUB	<address>	Subtract the contents of the given address from the ACC
	SUB	#n	Subtract the number n from the ACC
	STO	<address>	Store the contents of the ACC at the given address
<label>:		<data>	Gives a symbolic address <label> to the memory location with the contents <data>
# denotes a denary number, e.g. #123 <label> can be used in place of <address>			

The current contents of memory are:

Address	Contents
563	125
899	63

Write **assembly language** code, using **only** the given instruction set to:

- store the denary value 250 as labelled variable X
- store the value stored in location 563 as labelled variable Y
- add the value stored in variable X to the value stored in variable Y
- subtract the value stored in location 899 from the current value in the Accumulator
- store the result in variable Total.

Show the initialisation and values of the variables X, Y and Total in the table provided.

.....

.....

.....

.....

.....

.....

.....

.....

.....

Label	Content