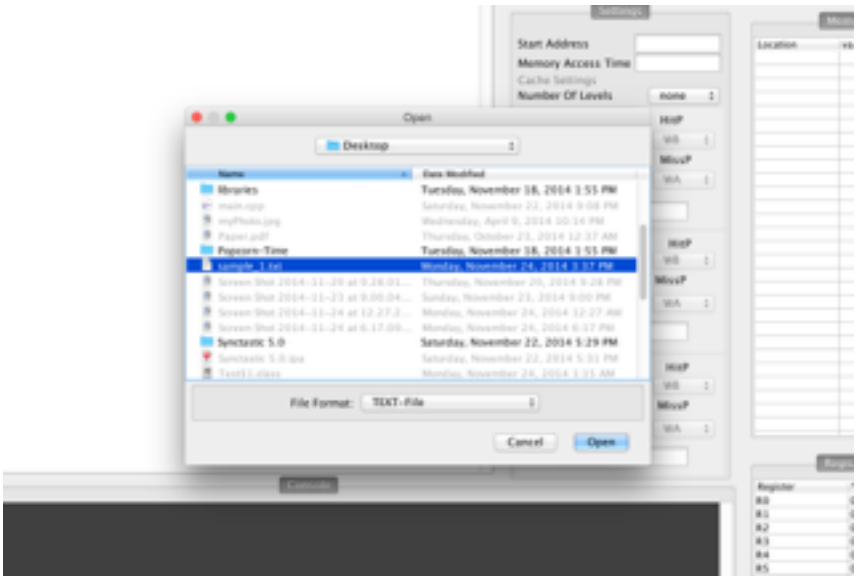
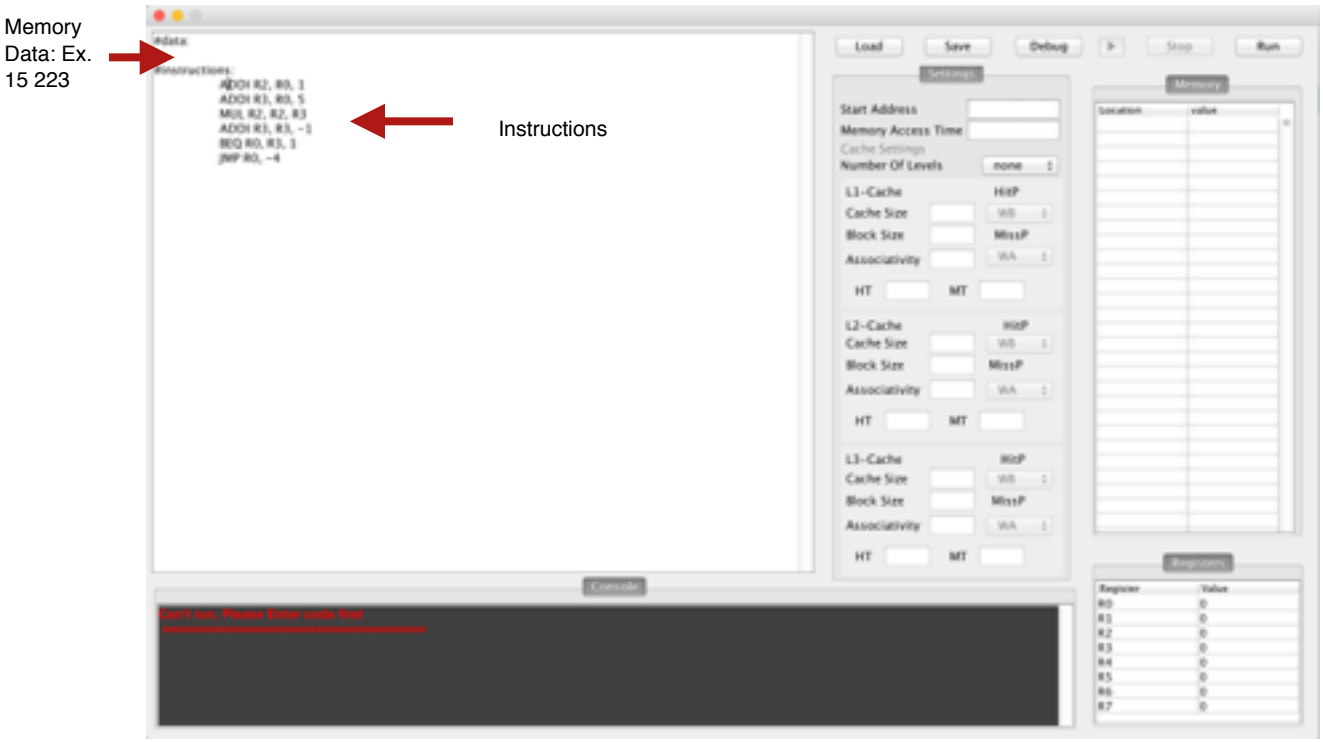
After Opening the program we will be promoted with figure(1). We have 2 ways to input a code, either by loading file or writing the code then save it.

1- We are going to load < sample_1.txt > by selecting load button as figure(2)

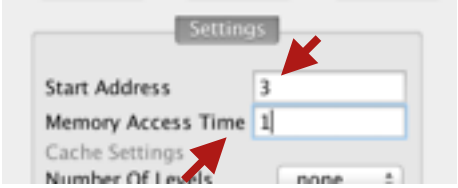
2- You will be promoted with a File browser, Choose the location of the file and select Open.





3- If there are any data you want add, in the editor after < #data: > enter key and the the value with a space separated (ex. 15 223). Any instructions should go after < #instructions: > .

4- Enter the starting address and the Memory Access Time is the Settings panel



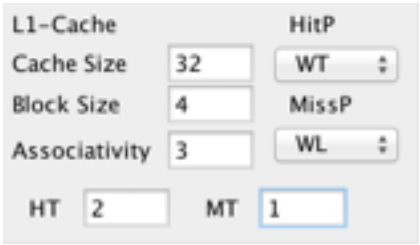
Testing level 1-cache (L1).

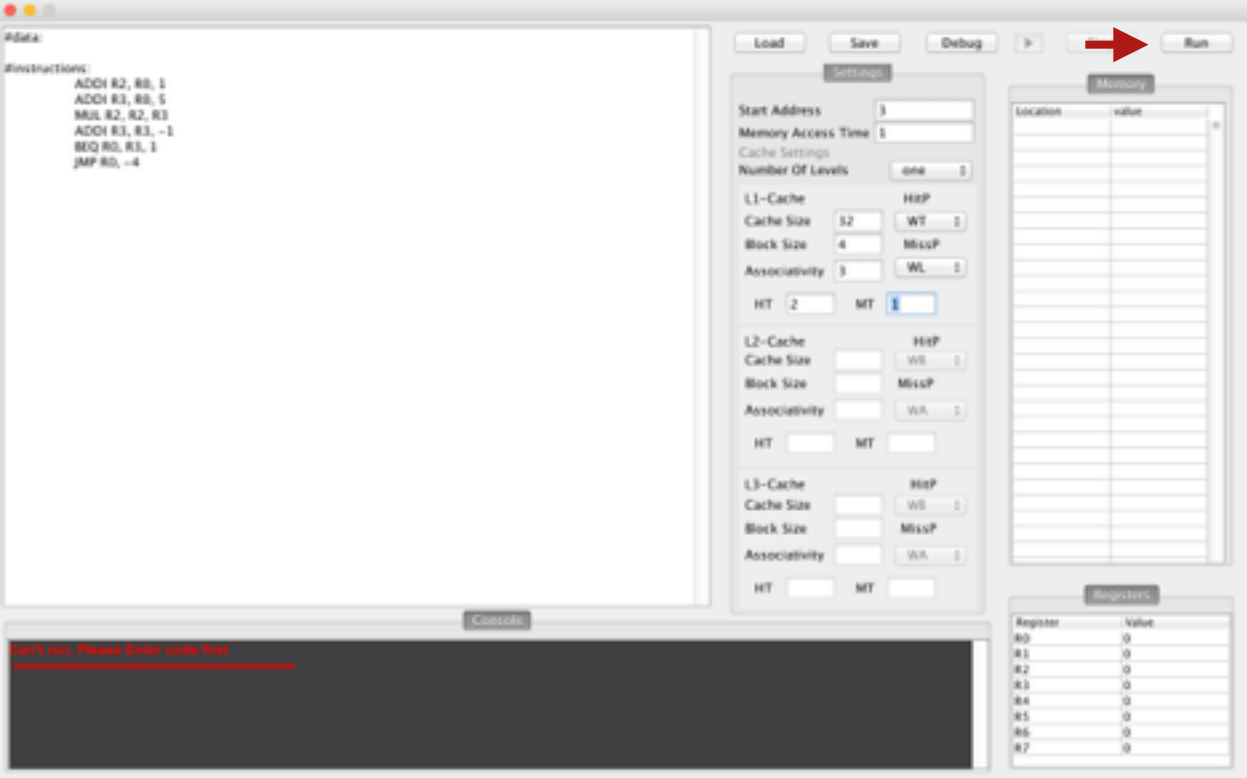
5- For choosing On Level Cache.. From the dropDown List of the Number of levels we select One .

6- Fill the Settings of the L1-Cache Level: Cache Size, Block Size, Associativity, the Hit/Miss policies and the Hit/Miss times.

Note that:

Hit Policies as following
{ WB: Write Back, WT: Write Through}
Miss Policies as following
{ WA: Write Around, WL: Write Allocate}
Associativity : ranges from 1…4

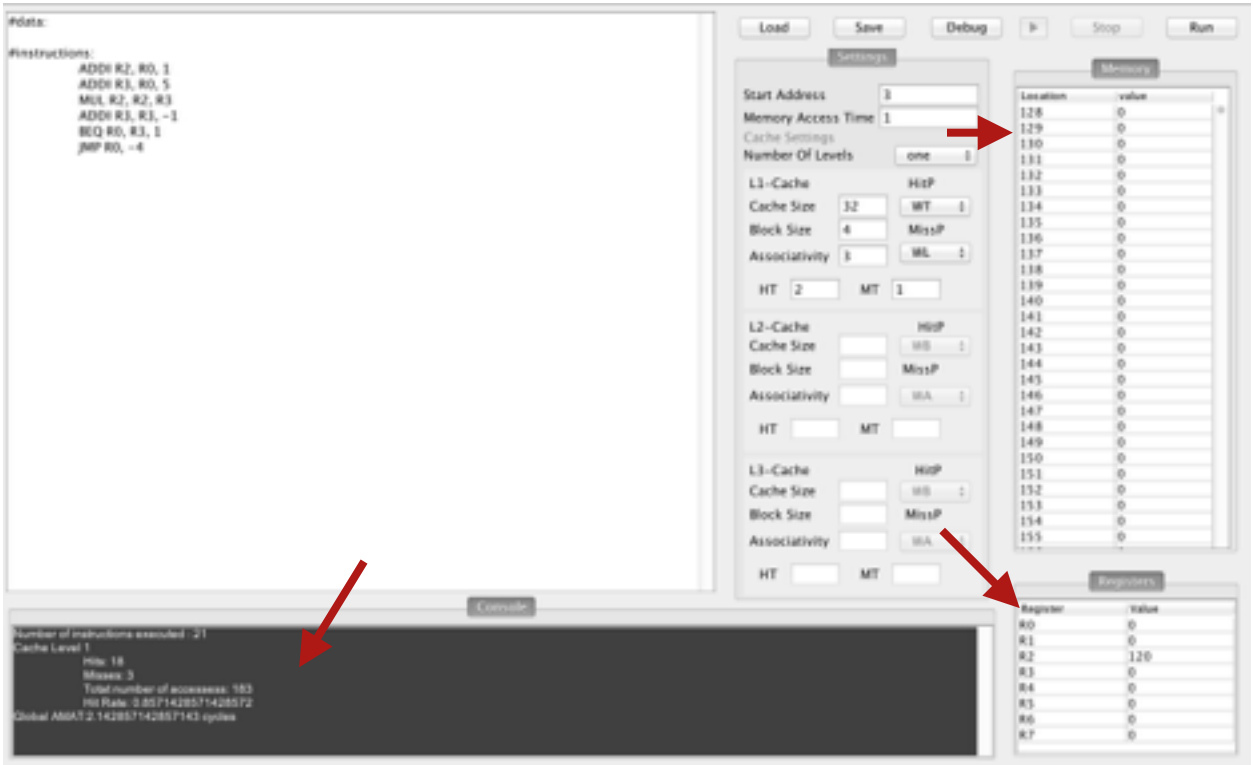




7- After all the setting has been set. now we can run the program.

Results of The level 1-cache (L1).

8- If every thing ran successfully u will promoted with figure <3> which contains the result in the console and the registers/Memory data , but if you forgot anything you will be given error/s of what is missing.



Analyzing the Results

9- The Console results will look like the following :

Cache Level

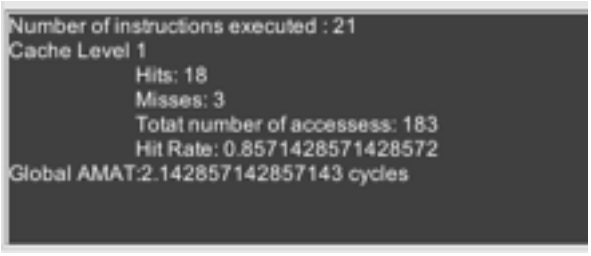
Hits: #

Misses: #

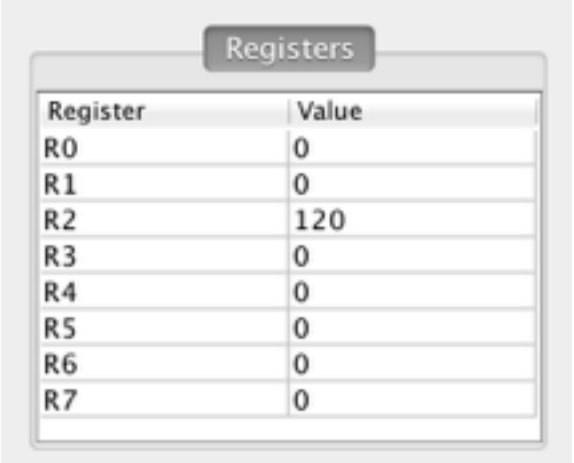
Total Number Of accesses: #

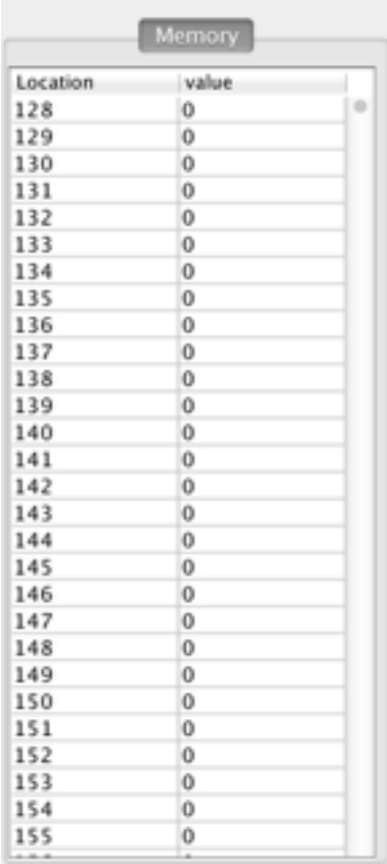
Hit Rate: #

The AMAT: #



10- The registers Data will contain all the data from R0…R7 that is been used in the program.





11- The Memory Will contain all the data that is stored in the memory.

Testing level 2-caches (L1- L2).

5- For choosing 2 Level Caches.. From the dropDown List of the Number of levels we select two .

6- Assuming the same configuration for the L1-Cache, we configure the L2-cache with smiler settings.

7- After all the setting has been set. now we can run the program.

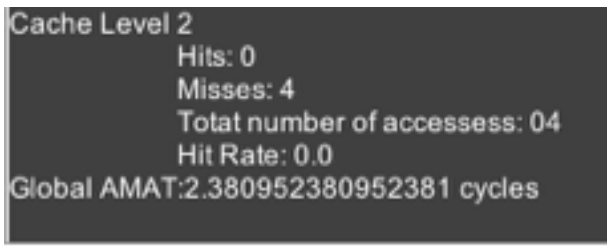
Results of The level 2-cache (L1-L2).

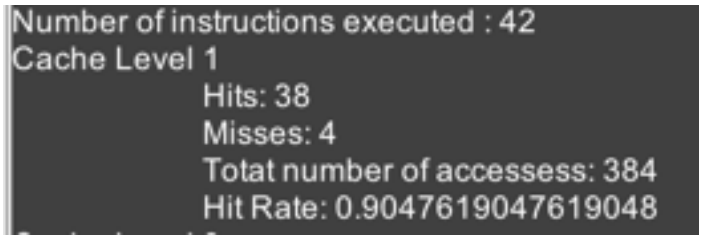
8- Same as the previous one.

Analyzing the Results

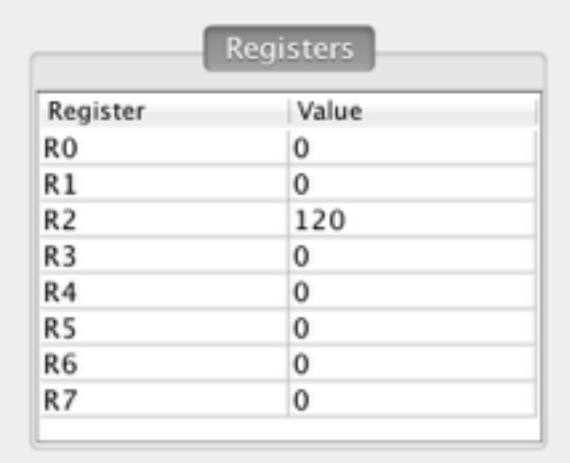
9- The Console results will look like the following :

Cache Level #
Hits: #
Misses: #
Total Number Of accesses: #
Hit Rate: #
The AMAT: #

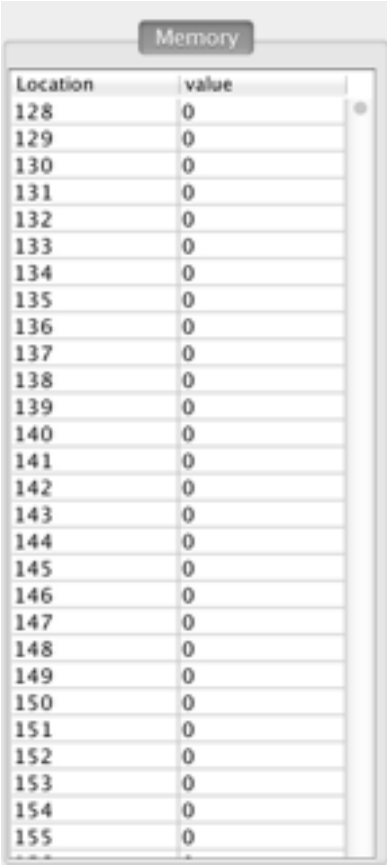




10- The registers Data will contain all the data from R0…R7 that is been used in the program.



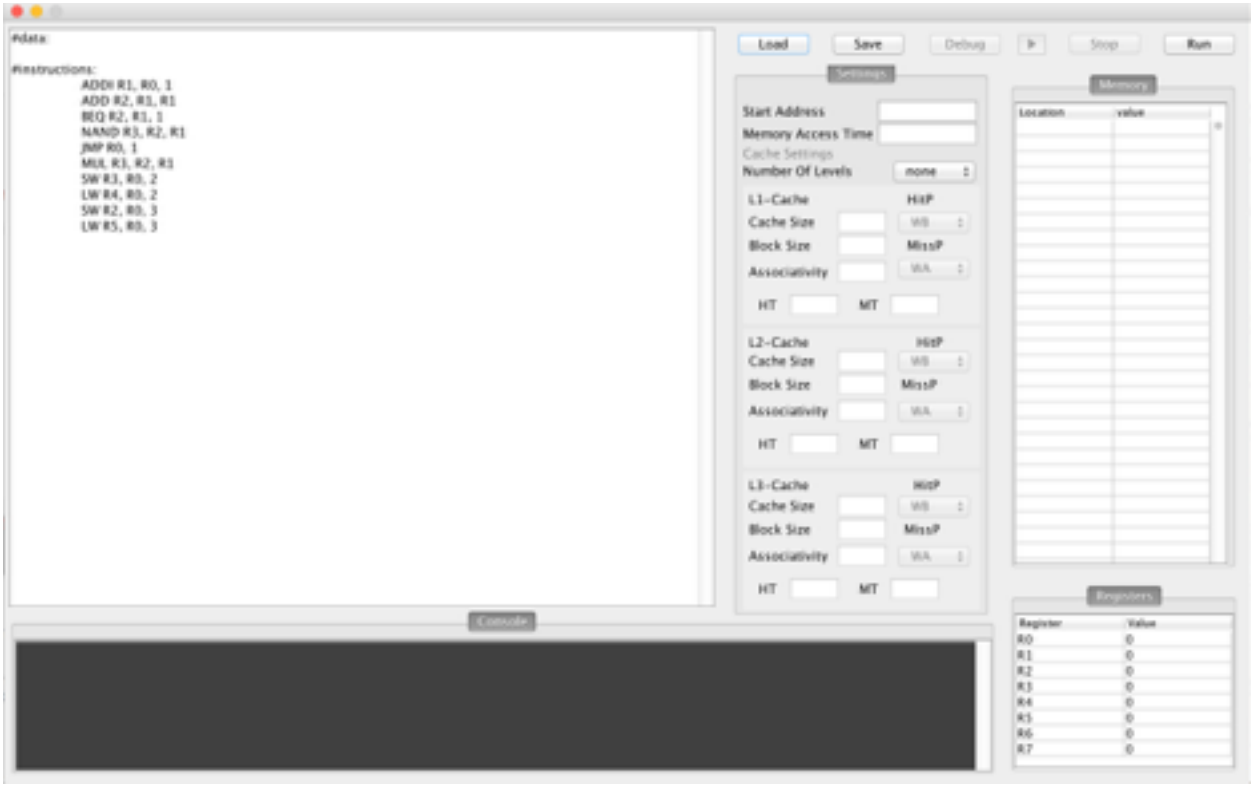
Register	Value
R0	0
R1	0
R2	120
R3	0
R4	0
R5	0
R6	0
R7	0



Location	value
128	0
129	0
130	0
131	0
132	0
133	0
134	0
135	0
136	0
137	0
138	0
139	0
140	0
141	0
142	0
143	0
144	0
145	0
146	0
147	0
148	0
149	0
150	0
151	0
152	0
153	0
154	0
155	0

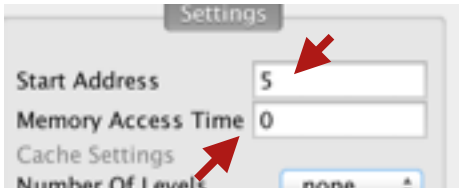
11- The Memory Will contain all the data that is stored in the memory.

Loading Sample_2.txt



Steps 1,2 & 3 are the same as the previous example.

4- Enter the starting address and the Memory Access Time is the Settings panel



Testing level 2-cache (L1-L2).

5- For choosing Two Levels of Caches.. From the dropDown List of the Number of levels we select Two .

6- Fill the Settings of the L1-Cache & L2- Cache with the required data.

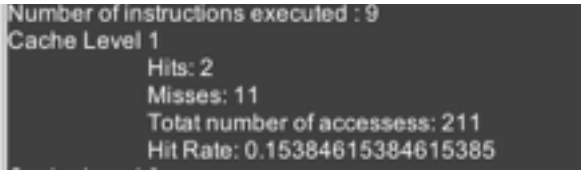
7- After all the setting has been set. now we can run the program.

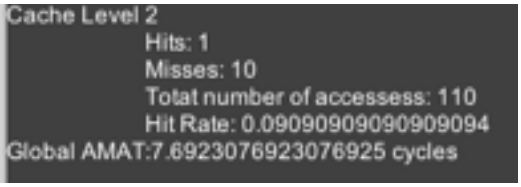
Results of The 2 levels 2-cache (L1-L2).

8- Same as the previous example.

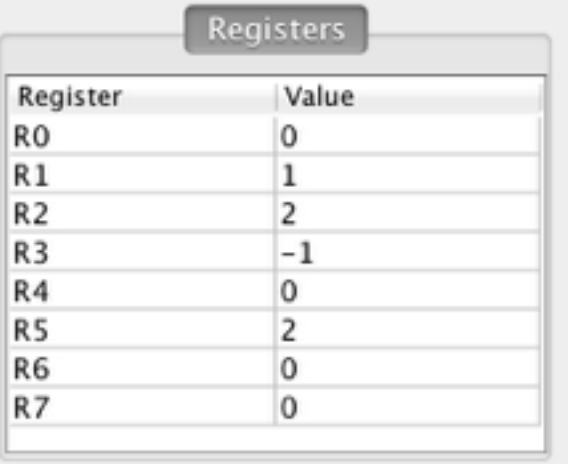
Analyzing the Results

9- The Console results will look like the following :





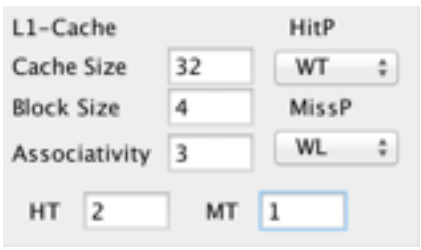
10- The registers Data will contain all the data from R0…R7 that is been used in the program.



Testing level 3-caches (L1- L2-L3).

5- For choosing 3 Levels of Caches.. From the dropDown List of the Number of levels we select Three .

6- Assuming the same configuration for the L1-Cache, we configure the L2-cache with smiler settings.



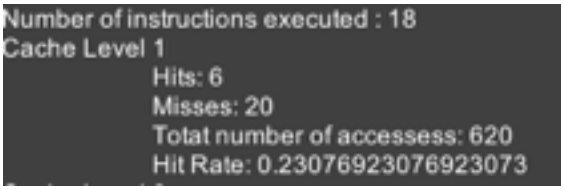
7- After all the setting has been set. now we can run the program.

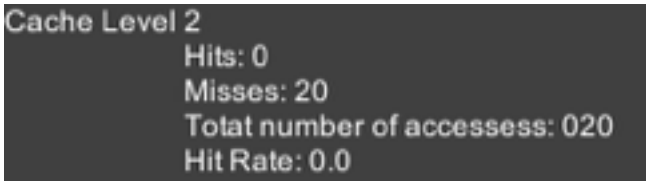
Results of The level 3-caches (L1-L2-L3).

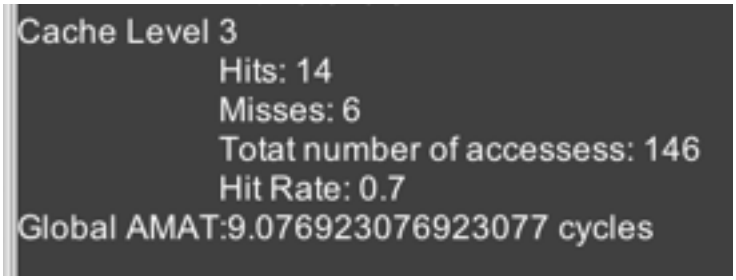
8- Same as the previous one.

Analyzing the Results

9- The Console results will look like the following :







10- The registers Data will contain all the data from R0…R7 that is been used in the program.

Registers

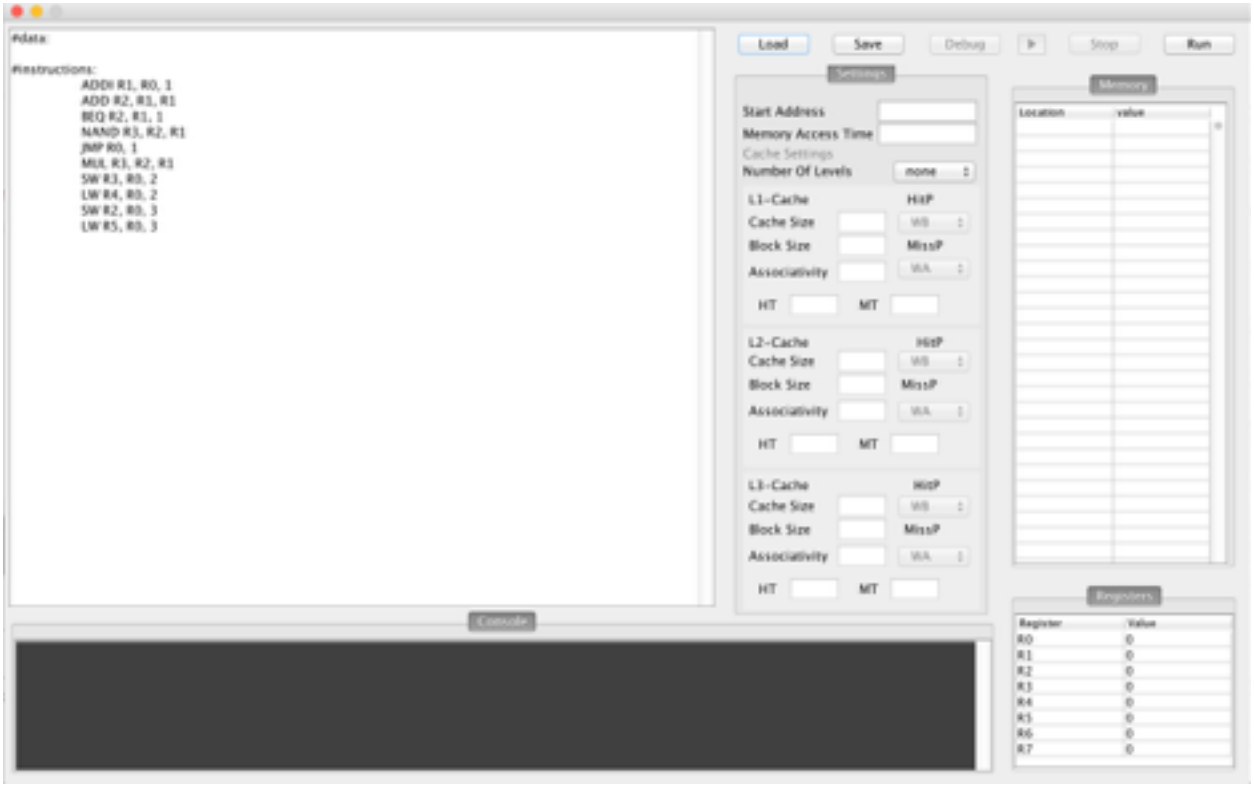
Register	Value
R0	0
R1	1
R2	2
R3	-1
R4	-1
R5	2
R6	0
R7	0

Memory

Location	value
100	0
101	0
102	-1
103	0
104	0
105	0
106	0
107	0
108	0
109	0
110	0
111	0
112	0
113	0
114	0
115	0
116	0
117	0
118	0
119	0
120	0
121	0
122	0
123	0
124	0
125	0
126	0
127	0

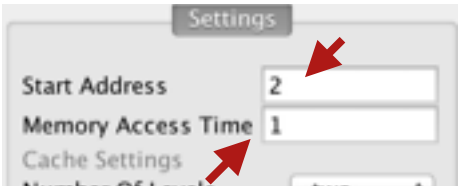
11- The Memory Will contain all the data that is stored in the memory.

Loading Sample_3.txt



Steps 1,2 & 3 are the same as the previous example.

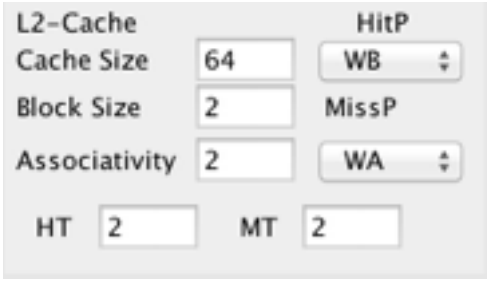
4- Enter the starting address and the Memory Access Time is the Settings panel



Testing level 2-cache (L1-L2).

5- For choosing Two Levels of Caches.. From the dropDown List of the Number of levels we select Two .

6- Fill the Settings of the L1-Cache & L2- Cache with the required data.



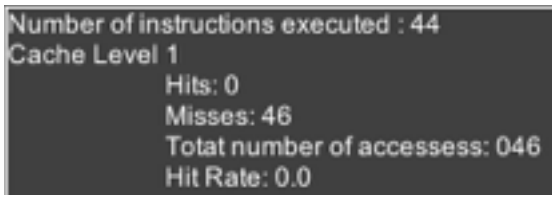
7- After all the setting has been set. now we can run the program.

Results of The 2 levels 2-cache (L1-L2).

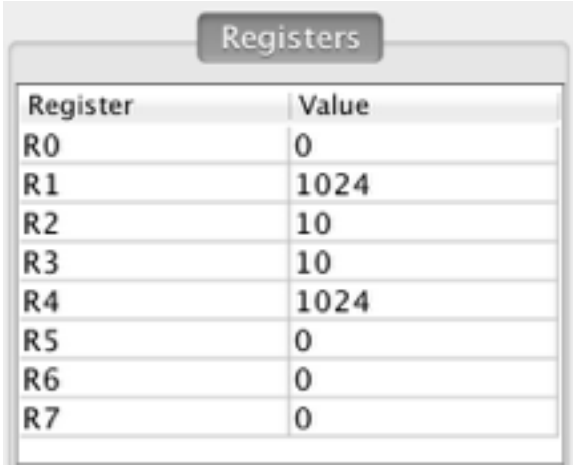
8- Same as the previous example.

Analyzing the Results

9- The Console results will look like the following :



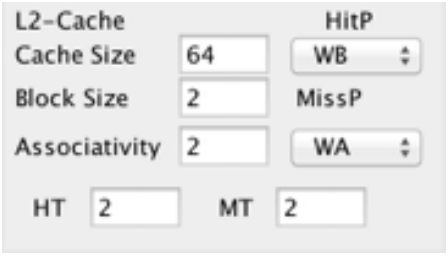
10- The registers Data will contain all the data from R0…R7 that is been used in the program.



Testing level 3-caches (L1- L2-L3).

5- For choosing 3 Levels of Caches.. From the dropDown List of the Number of levels we select Three .

6- Assuming the same configuration for the L1-Cache, we configure the L2-cache with smiler settings.



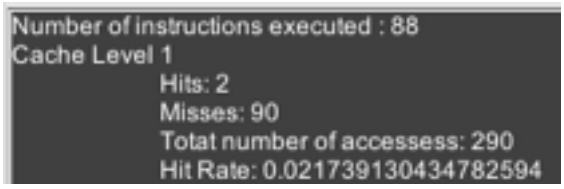
7- After all the setting has been set. now we can run the program.

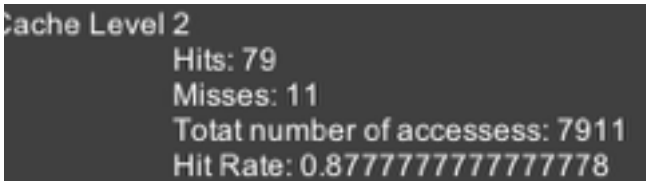
Results of The level 3-caches (L1-L2-L3).

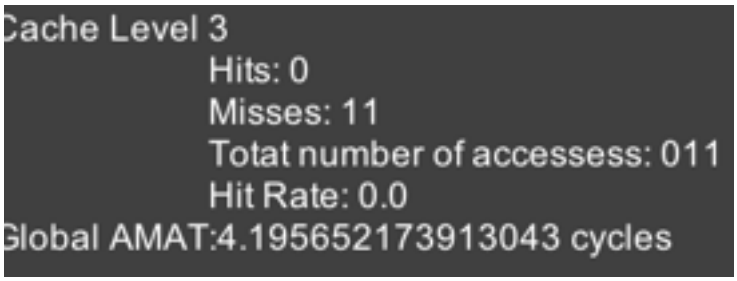
8- Same as the previous one.

Analyzing the Results

9- The Console results will look like the following :







Registers

Register	Value
R0	0
R1	1024
R2	10
R3	10
R4	1024
R5	0
R6	0
R7	0

10- The registers Data will contain all the data from R0…R7 that is been used in the program.

Memory

Location	value
100	0
101	0
102	0
103	0
104	1024
105	0
106	0
107	0
108	0
109	0
110	0
111	0
112	0
113	0
114	0
115	0
116	0
117	0
118	0
119	0
120	0
121	0
122	0
123	0
124	0
125	0
126	0
127	0

11- The Memory Will contain all the data that is stored in the memory.