

Programming Assignment #2

Assignment Description

(100 points = 70% of demonstration + 30% of report)

Write a simple SIC simulator that reads a standard SIC object program into memory.

Basic Functionalities

(70 points for demonstrating the functionality)

The simulator takes two commands:

- (a) **load *ObjectFileName***: loads a simple SIC simulator that reads a standard SIC object program into assigned location in memory.
- (b) **show**: shows the contents in the memory location for the loaded object program.

Advanced Functionalities

(10 more points for demonstrating the functionality)

In addition to basic functionalities, the simulator takes two more commands:

- (c) **unload**: releases the data structure for the loaded object program.
- (d) **exit**: exits the simulator successfully.

Moreover, the **load** command will show an error message if there is already an object program loaded in memory.

Full Functionalities

(20 more points for demonstrating the functionality)

In addition to advanced functionalities, the simulator takes one more command:

- (e) **run**: simulates the execution of the loaded object program, and shows the contents of all registers when it ends.

Goals

1. Get familiar with C programming language.
2. Learn to use the I/O facilities and library functions provided by standard C.
3. Get experience with the system-level programming.
4. Get experience with separate compilation, make utility, and C debugger.

Guideline:

1. You have to demonstrate your program in person.
2. You will get 15% bonus if you succeed in demonstrating your program on December 23rd, 2022. Official due for demonstrating program is January 6th, 2023. After that, 15% penalty will be given for lateness. More precisely, if you get X in demonstration, and Y for the report:
 - (12/23) In-class demonstration = $X * 70\% * 115\% + Y * 30\%$
 - (1/6) Your score = $X * 70\% + Y * 30\%$
 - Late = $(X * 70\% + Y * 30\%) * 85\%$
3. Your report has to include the following elements:
 - 1) A coverpage.
 - 2) The problem description.
 - 3) Highlight of the way you write the program.
 - 4) The program listing.
 - 5) Test run results.
 - 6) Discussion.

Programming Assignment #2 Addendum

Example Object Programs

1)

HCOPY 00100000107A

T0010001E1410334820390010362810303010154820613C100300102A0C103900102D

T00101E150C10364820610810334C0000454F46000003000000

T0020391E041030001030E0205D30203FD8205D2810303020575490392C205E38203F

T0020571C1010364C0000F1001000041030E02079302064509039DC20792C1036

T002073073820644C000005

E001000

2)

MATH 00200000002D

T0020001E00201E1820241C201B0C20270020211820241C201B0C202A4C0000000001

T00201E09000005000007000003

E002000

Format of Output

(Similar to the following figure)

Memory address	Contents			
0000	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
⋮	⋮	⋮	⋮	⋮
3FF0	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
4000
4010
4020	03201D77	1040C705	0014.....
4030
4040
4050	00412600	00080040	51000004
4060	000083.
4070
4080
4090031040	40772027
40A0	05100014
40B0
40C0
40D000	41260000	08004051	00000400
40E0	0083.
40F00310	40407710
4100	40C70510	0014.....
4110
4120	00412600	00080040	51000004
4130	000083xx	xxxxxxxx	xxxxxxxx	xxxxxxxx
4140	xxxxxxxx	xxxxxxxx	xxxxxxxx	xxxxxxxx
⋮	⋮	⋮	⋮	⋮

← PROGA

← PROGB

← PROGC

When demonstrating

1. Enter your program and a prompt.
2. Load test1.obj 1000
3. Show 1000
4. Run 1000
5. Show 1000
6. Unload
7. Load test2.obj 2000
8. Show 2000
9. Run 2000
10. Show 2000
11. Exit