Project Report

System Software

CSN 252

Textual Draw

Members

● Abhinav Saini	19114001
● Jatin	19114037
● Jinendra Verma	19114038
● Rahul Agrawal	19114069
● Retiwala Harsh Prakash	19114072
● Ritikesh Ramdas Deshbhratar	19114073

Problem Statement

Draw on a textual screen by reading the commands from standard input. Type a command (or more commands) to standard input and press enter to execute them. Implement the following commands.

- h : displays help on stdout
- w, a, s, d : up, left, down, right
- f : changes drawing symbol to next typed character
- c : clears the screen and returns to center
- p : fills the screen with drawing symbol
- q : halt

Examples:

- 'aaaa wwww dddd ssss' -> draws a 4x4 square
- 'f.' -> change the drawing symbol to '.'
- 'f-dddf df-dddf df-ddd' -> draws a dashed line '--- --- ---'

Explanation

In this problem we have to construct a white board type system in which a user can draw any object or shape using W/A/S/D keys on the keyboard for drawing up, left, down and right respectively.

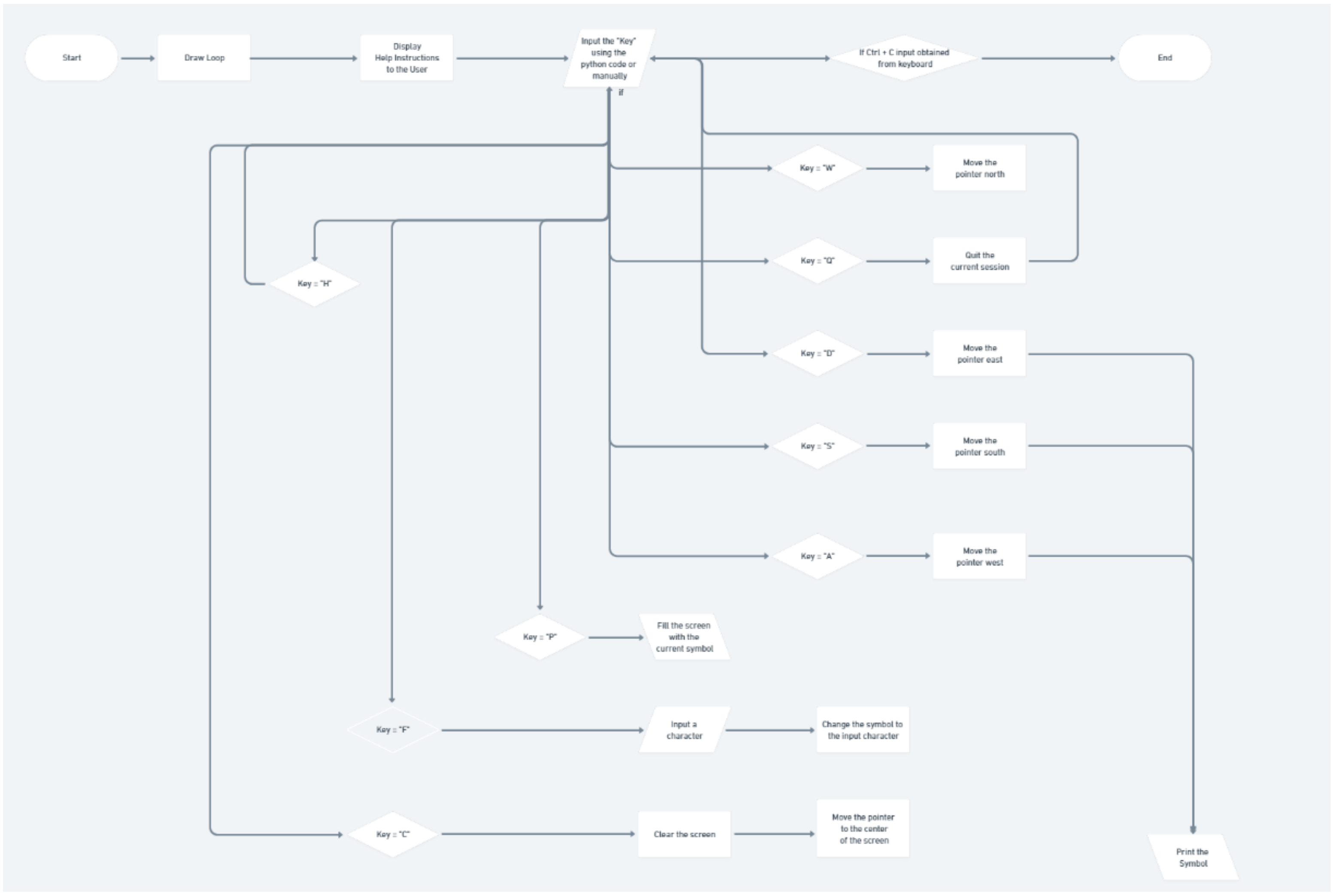
Initially the drawing coordinate will start from center then it will shift according to the user input. In Addition, a user can change drawing symbols like ,$,^,*,&..... etc. using the ‘F’ key. ‘C’ will clear the screen, ‘P’ will fill it with symbols and ‘Q’ will be used for halting.

Routines and SubRoutines

- Reset: Our Program starts with this subroutine.Whenever this is called,first screenColumns is loaded into accumulator and then it is divided by two and then multiplied with screenRow.Now the calculated value is stored in register X.Then,help subroutine is called and helpText is printed.Now the printMousePtr is called which basically print a mouse pointer at the value stored in X.The value stored in X is nothing basically the center of the screen.
- Input: This basically reads the command entered by the user and matches it to the subroutine assigned to the command.
- Switch: This routine is called when the user enters f and then it reads the next character entered along with x and stores it in the variable symbol.
- printMousePtr: This routine prints the mousePtr on the screen.This then calls the input subroutine.
- fillScreen: The symbol is loaded into the accumulator and then it calls the subroutine screenFill which fills the whole screen with the value in the accumulator and then reset is called.
- clearScreen: This first calls the subroutine screenClear and then reset is called,which sets the mousePtr at the center.
- screenClear: This routine is called by the clearScreen method,which basically loads the ascii value of space character in the accumulator and then writeLastA subroutine is called,which fills the screen with space character that is it clears the screen.
- screenFill: This routine is called by the fillScreen method,which basically stores the value in the accumulator to tempA and then the writeLastA subroutine is called,which fills the screen with the current symbol.
- help: This subroutine calculates the length of the help text and stores it in helpLength.
- helpAndWrite: This subroutine prints the helpText in the stdout.It run a loop of the length helpLength and it print character wise and then when the looping variable is equal to helpLength,this loop is stopped and restoreX is called,which restores the value of X using tempX and reset accumulator to zero and then printMousePtr on the textual screen.
- restoreX: This subroutine restores the value of the register X using tempX and reset the accumulator to zero and then printMousePtr on the textual screen.

- writeLastA: This routine is called to store the last bit of A in the screen variable.This is called when we have to either clear the whole screen or print the whole screen.So,initially the value stored in the register X is temporarily stored to tempX. Then register X is loaded with 0 and loop of length screenLength is called and it stores the current bit of A in screen and then as the loop finishes,return is called,which restores the values of the register A and X.
- return: This stores the values of tempX and tempA in the register X and Accumulator respectively.Now,here this subroutine is returned.
- halt: This subroutine when called,stops the program.
- moveUp:This print the symbol at the mouseptr and move the mouse pointer by 1 unit in the positive y direction.If the mouse pointer is already at the top of the textual screen, it changes its location to the bottom of the screen, keeping the location in the x direction same.
- moveDown:This print the symbol at the mouseptr and move the mouse pointer by 1 unit in the negative y direction.If the mouse pointer is already at the bottom of the textual screen, it changes its location to the top of the screen, keeping the location in the x direction same.
- moveLeft:This print the symbol at the mouseptr and move the mouse pointer by 1 unit in the negative x direction.If the mouse pointer is already at the left of the textual screen, it changes its location to the right of the screen, keeping the location in the y direction same.
- moveRight:This print the symbol at the mouseptr and move the mouse pointer by 1 unit in the positive x direction.If the mouse pointer is already at the right of the textual screen, it changes its location to the left of the screen, keeping the location in the y direction same.
- addLength: This subroutine mainly is responsible for changing the position of the mouse pointer when moveUp is called.
- subtractLength: This subroutine mainly is responsible for changing the position of the mouse pointer when moveDown is called.
- subtractScreenColoumns: This subroutine mainly is responsible for changing the position of the mouse pointer when moveRight is called.
- addScreenColoumns: This subroutine mainly is responsible for changing the position of the mouse pointer when moveLeft is called.

Flowchart



Flowchart link - https://whimsical.com/37c1zmNWK4yYVc9zuQsSN4

Result

To prevent manually entering all the symbols in the correct positions we wrote a python script to convert a given text file into a command block which then can be directly sent in the command line to the simulator. The text file includes the ascii symbol art which we want to show on the textual screen.

As the default size of the SicTools Textual screen is 25x80 pixels, to accommodate our text we changed the source code for the tool to be 100x100 pixels and then made our jar. And then the counter is set to start from the origin of the screen using python.

For changing the screen size,we made changes to `/SicTools/src/sic/sim/addons/TextualScreen.java` and then made the jar.

Python code

```
import string

f = open("Text", 'r')
symbols = string.ascii_lowercase + string.ascii_uppercase +
string.punctuation + ' '
text = f.readlines()

commands = ""
commands += "f \n"
for j in range(50):
      commands += "w\n"

def output(l):
      s = ""
      for i in l:
            if i in symbols:
                  s += "f{0}".format(i) + "\n"
                  s += "d\n"
      s += "s\n"
      s += "f \nw\n"
      return s

o = open("code", 'w')

for l in text:
      commands += output(l)

o.write(commands)
```

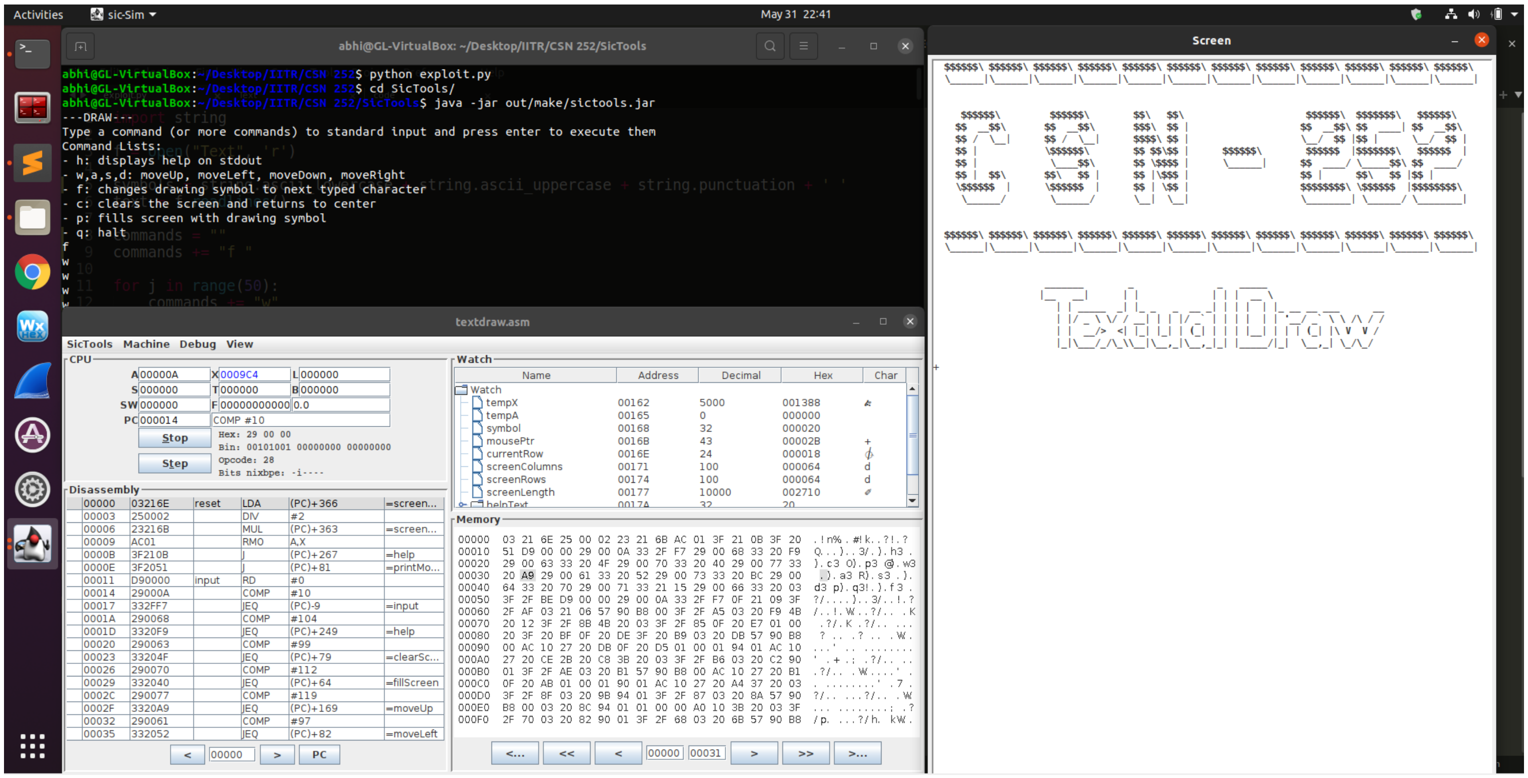
Text file (input)

```
 $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\
 \______|\______|\______|\______|\______|\______|\______|\______|\______|\______|\______|\______|


    $$$$$$\         $$$$$$\        $$\   $$\                      $$$$$$\  $$$$$$$\   $$$$$$\
   $$  __$$\       $$  __$$\       $$$\  $$ |                    $$  __$$\ $$  ____| $$  __$$\
   $$ /  \__|      $$ /  \__|      $$$$\ $$ |                    \__/  $$ |$$ |      \__/  $$ |
   $$ |            \$$$$$$\        $$ $$\$$ |     $$$$$$\        $$$$$$  |$$$$$$$\   $$$$$$  |
   $$ |             \____$$\       $$ \$$$$ |     \______|      $$  ____/ \_____$$\ $$  ____/
   $$ |  $$\       $$\   $$ |      $$ |\$$$ |                    $$ |      $$\   $$ |$$ |
   \$$$$$$  |      \$$$$$$  |      $$ | \$$ |                    $$$$$$$$\ \$$$$$$  |$$$$$$$$\
    \______/        \______/       \__|  \__|                    \________| \______/ \________|


 $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\ $$$$$$\
 \______|\______|\______|\______|\______|\______|\______|\______|\______|\______|\______|\______|


                   _______        _               _   _____
                  |__   __|      | |             | | |  __ \
                     | | _____  _| |_ _   _  __ _| | | |  | |_ __ __ ___      __
                     | |/ _ \ \/ / __| | | |/ _` | | | |  | | '__/ _` \ \ /\ / /
                     | |  __/>  <| |_| |_| | (_| | | | |__| | | | (_| |\ V  V /
                     |_|\___/_/\_\\__|\__,_|\__,_|_| |_____/|_|  \__,_| \_/\_/
```

Commands (output)

```
f wwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwwf df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df
df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df
df$df$df$df$df$df$df\df df df dsf wf df
df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df
\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df df dsf wf df df df df df df df df df df df df df
df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df
df df df df df df df df df df df df df df df df df df df df df df df df df df df df df dsf wf df df df df df df df df df df df df df df df df df df df df df df df df df
df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df
df df df df df df df df df df df df df df df df df dsf wf df df df df df$df$df$df$df$df$df\df df df df df df df df df df$df$df$df$df$df$df\df df df df df df df df
df$df$df\df df df df$df$df\df df df df df df df df df df df df df df df df df df df df df df df df$df$df$df$df$df$df\df df df$df$df$df$df$df$df$df\df df df
df$df$df$df$df$df$df\df df df df df df dsf wf df df df df$df$df df df_df_df$df$df\df df df df df df df df$df$df df df_df_df$df$df\df df df df df df df df$df$df$df\df df
df$df$df df|df df df df df df df df df df df df df df df df df df df df df$df$df df df_df_df$df$df\df df$df$df df df_df_df_df_df|df df$df$df df df_df_df$df$df\df df df df
df dsf wf df df df df$df$df df/df df df\df_df_df|df df df df df df df$df$df df/df df df\df_df_df|df df df df df df df$df$df$df$df\df df$df$df df|df df df df df df df df
df df df df df df df df df df df df df\df_df_df/df df df$df$df df|df$df$df df|df df df df df df df\df_df_df/df df df$df$df df|df df df df dsf wf df df df df$df$df df|df
df df df df df df df df df df df df\df$df$df$df$df$df$df\df df df df df df df df df$df$df df$df$df\df$df$df df|df df df df df df df$df$df$df$df$df$df\df df df df df df df
df df$df$df$df$df$df$df df df|df$df$df$df$df$df$df$df\df df df df$df$df$df$df$df$df df df|df df df df dsf wf df df df df$df$df df|df df df df df df df df df df df df df
df\df_df_df_df_df$df$df\df df df df df df df df$df$df df\df$df$df$df$df df|df df df df df df df\df_df_df_df_df_df_df|df df df df df df df$df$df df df_df_df_df_df/df
df\df_df_df_df_df_df$df$df\df df$df$df df df_df_df_df_df/df df df df df dsf wf df df df df$df$df df|df df df$df$df\df df df df df df df df$df$df\df df df df$df$df df|df
df df df df df df$df$df df|df\df$df$df$df df|df df df df df df df df df df df df df df df df df df df df df$df$df df|df df df df df df df$df$df\df df df df$df$df
df|df$df$df df|df df df df df df df df df df dsf wf df df df df\df$df$df$df$df$df$df df df|df df df df df df df\df$df$df$df$df$df$df df df|df df df df df df df$df$df
df|df df\df$df$df df|df df df df df df df df df df df df df df df df df df df df df$df$df$df$df$df$df$df$df\df df\df$df$df$df$df$df$df df df|df$df$df$df$df$df$df$df$df\df
df df df df dsf wf df df df df df\df_df_df_df_df_df_df/df df df df df df df df df\df_df_df_df_df_df_df/df df df df df df df df\df_df_df|df df df\df_df_df|df df df df df
df df df df df df df df df df df df df df df df\df_df_df_df_df_df_df_df_df|df df\df_df_df_df_df_df_df/df df\df_df_df_df_df_df_df_df_df|df df df df dsf wf df df df df df
df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df
df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df dsf wf df df df df df df df df df df df df df df df df df
df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df
df df df df df df df df df df df df df df df df df df df df df df df df df dsf wf df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df
df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df
df$df$df$df$df$df$df\df df$df$df$df$df$df$df\df df df dsf wf df
df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df
\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df\df_df_df_df_df_df_df|df df dsf wf df df df df df df df df df df df df df
df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df
df df df df df df df df df df df df df df df df df df df df df df df df df df df df df dsf wf df df df df df df df df df df df df df df df df df df df df df df df df df
df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df
df df df df df df df df df df df df df df df df df dsf wf df df df df df df df df df df df df df df df df df df df df_df_df_df_df_df_df_df df df df df df df df df_df df
df df df df df df df df df df df df df df_df df df df_df_df_df_df_df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df
df df df df df dsf wf df df df df df df df df df df df df df df df df df df df|df_df_df df df df_df_df|df df df df df df|df df|df df df df df df df df df df df df
df|df df|df df|df df df_df_df df\df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df dsf wf df df df df
df df df df df df df df df df df df df df df df df|df df|df df_df_df_df_df_df df df_df|df df|df_df df_df df df df_df df df_df_df df_df|df df|df df|df df|df df df|df
df|df_df df_df_df df_df_df df_df_df_df df df df df df df_df_df df df df df df df df df df df df df df df df df df df dsf wf df df df df df df df df df df df df df df df
df df df df df df df|df df|df/df df_df df\df df\df/df df/df df_df_df|df df|df df|df df|df/df df_df`df df|df df|df df|df df|df df df|df df|df df'df_df_df/df df_df`df df\df
df\df df/df\df df/df df/df df df df df df df df df df df df df df df df df df df dsf wf df df df df df df df df df df df df df df df df df df df df df df|df df|df df
df_df_df/df>df df df<df|df df|df_df|df df|df_df|df df|df df(df_df|df df|df df|df df|df df|df_df_df|df df|df df|df df|df df(df_df|df df|df\df dfVdf df dfVdf df/df df df df
df df df df df df df df df df df df df df df df dsf wf df df df df df df df df df df df df df df df df df df df df df
df|df_df|df\df_df_df_df/df_df/df\df_df\df\df_df_df|df\df_df_df,df_df|df\df_df_df,df_df|df_df|df df|df_df_df_df_df_df/df|df_df|df df df\df_df_df,df_df|df
df\df_df/df\df_df/df df df df df df df df df df df df df df df df df df df df df dsf wf df df df df df df df df df df df df df df df df df df df df df df df df df df df
df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df df
df df df df df df df df df df df df df df df dsf w
```

Output



```
abhi@GL-VirtualBox:~/Desktop/IITR/CSN 252$ python exploit.py
abhi@GL-VirtualBox:~/Desktop/IITR/CSN 252$ cd SicTools/
abhi@GL-VirtualBox:~/Desktop/IITR/CSN 252/SicTools$ java -jar out/make/sictools.jar
---DRAW---
Type a command (or more commands) to standard input and press enter to execute them
Command Lists:
- h: displays help on stdout
- w,a,s,d: moveUp, moveLeft, moveDown, moveRight
- f: changes drawing symbol to next typed character
- c: clears the screen and returns to center
- p: fills screen with drawing symbol
- q: halt
f
w
w
w
w
```

textdraw.asm

CPU

```
A  00000A   X 0009C4        L 000000
S  000000   T 000000        B 000000
SW 000000   F 00000000000   0.0
PC 000014   COMP #10
            Hex: 29 00 00
            Bin: 00101001 00000000 00000000
            Opcode: 28
            Bits nixbpe: -i----
```

Disassembly

00000	03216E	reset	LDA	(PC)+366	=screen...
00003	250002		DIV	#2	
00006	23216B		MUL	(PC)+363	=screen...
00009	AC01		RMO	A,X	
0000B	3F210B		J	(PC)+267	=help
0000E	3F2051		J	(PC)+81	=printMo...
00011	D90000	input	RD	#0	
00014	29000A		COMP	#10	
00017	332FF7		JEQ	(PC)-9	=input
0001A	290068		COMP	#104	
0001D	3320F9		JEQ	(PC)+249	=help
00020	290063		COMP	#99	
00023	33204F		JEQ	(PC)+79	=clearSc...
00026	290070		COMP	#112	
00029	332040		JEQ	(PC)+64	=fillScreen
0002C	290077		COMP	#119	
0002F	3320A9		JEQ	(PC)+169	=moveUp
00032	290061		COMP	#97	
00035	332052		JEQ	(PC)+82	=moveLeft

Watch

Name	Address	Decimal	Hex	Char
Watch				
tempX	00162	5000	001388	
tempA	00165	0	000000	
symbol	00168	32	000020	
mousePtr	0016B	43	00002B	+
currentRow	0016E	24	000018	
screenColumns	00171	100	000064	d
screenRows	00174	100	000064	d
screenLength	00177	10000	002710	
helpText	0017A	32	20	

Memory

```
00000  03 21 6E 25 00 02 23 21 6B AC 01 3F 21 0B 3F 20  .!n%.#!k..?!.?
00010  51 D9 00 00 29 00 0A 33 2F F7 29 00 68 33 20 F9  Q...)..3/.).h3 .
00020  29 00 63 33 20 4F 29 00 70 33 20 40 29 00 77 33  ).c3 O).p3 @).w3
00030  20 A9 29 00 61 33 20 52 29 00 73 33 20 BC 29 00   .).a3 R).s3 .).
00040  64 33 20 70 29 00 71 33 21 15 29 00 66 33 20 03  d3 p).q3!.).f3 .
00050  3F 2F BE D9 00 00 29 00 0A 33 2F F7 0F 21 09 3F  ?/....)..3/..!.?
00060  2F AF 03 21 06 57 90 B8 00 3F 2F A5 03 20 F9 4B  /..!.W..?/.. .K
00070  20 12 3F 2F 8B 4B 20 03 3F 2F 85 0F 20 E7 01 00   .?/.K .?/.. ...
00080  20 3F 20 BF 0F 20 DE 3F 20 B9 03 20 DB 57 90 B8   ? .. .? .. .W.
00090  00 AC 10 27 20 DB 0F 20 D5 01 00 01 94 01 AC 10  ...' .. ........
000A0  27 20 CE 2B 20 C8 3B 20 03 3F 2F B6 03 20 C2 90  ' .+ .; .?/.. ..
000B0  01 3F 2F AE 03 20 B1 57 90 B8 00 AC 10 27 20 B1  .?/.. .W....' .
000C0  0F 20 AB 01 00 01 90 01 AC 10 27 20 A4 37 20 03  . ........' .7 .
000D0  3F 2F 8F 03 20 9B 94 01 3F 2F 87 03 20 8A 57 90  ?/.. ...?/.. .W
000E0  B8 00 03 20 8C 94 01 01 00 00 A0 10 3B 20 03 3F  ... ........; .?
000F0  2F 70 03 20 82 90 01 3F 2F 68 03 20 6B 57 90 B8  /p. ...?/h. kW.
```