Raul Aguilar

CS 220

October 4, 2019

Homework 5: Machine Language

Mult.asm

// check if R0 or R1 is 0, if so skip to end

// otherwise continue

@2 // reset answer

M=0

@0 // A = 0

D=M // check if @0 == 0 (D = data in RAM[0)]

@END

D;JEQ // skip to end if zero

@1 // A = 1

D=M // check if RAM[1] == 0 (D = data in RAM[1])

@END

D;JEQ // skip to end if 0

(LOOP)

@2 // set A to where product is being stored (A = 2)

D=M // take the product already calculated (D = data in RAM[2])

@0 // go to R0

D=D+M // and add it to the product (M+D)

@2

M=D // store the product back into R2

@1 // take value in R1

MD=M-1 // and decrement how many times we need to add R0 to R2 (and store it in D register to compare)

@END

D;JEQ // check if R1 is at 0 and end loop

@LOOP

D;JGT // otherwise continue multiplication

(END) // infinite loop to end program

@END

0;JMP

A screenshot of a video game

Description automatically generated

Div.asm

// check if R0 or R1 is 0, if so skip to end

// otherwise continue

@2 // reset answer

M=0

@0 // A = 0

D=M // check if @0 == 0 (D = data in RAM[0])

@END

D;JEQ // skip to end if zero

@1 // A = 1

D=M // check if @1 == 0 (D = data in RAM[1])

@END

D;JEQ // skip to end if 0

(LOOP)

@0 // set A to where dividend is being stored (A = R0)

D=M // take the dividend (D = RAM[0])

@1 // go to stored divisor

D=D-M // and sub it from the dividend (M-D)

@0

M=D // store the difference back into R0

@END

D;JLT // if difference was < 0 then end division

@2

M=M+1 // otherwise increment how many times the divisor goes into the dividend/difference

@LOOP

D;JGT // and do it again

(END) // infinite loop to end program

@END

0;JMP

A screenshot of a social media post

Description automatically generated

Fill.asm

(START)

@counter

M=0

(LOOP)

@KBD // check if key is being pressed

D=M

@WHITE

D;JEQ // if not go to white (D == 0)

@BLACK

D;JGT // if true then go to black (D > 0)

(WHITE)

@counter // go to counter

D=M // get counter value

@SCREEN // A = 16384 first pixel of screen

A=D+A // A = RAM[SCREEN+counter] to go to current pixel

M=0 // paint it white

@counter // increment counter

MD=M+1

@8192 // check if at final pixel

D=D-A // 8192 - counter

@START

D;JEQ // reset if at final pixel

@LOOP

D;JLT // keep painting white

(BLACK)

@counter

D=M // get counter value

@SCREEN

A=D+A // add it to screen to get current position

M=-1 // paint it black

@counter

MD=M+1 // increment counter

@8192

D=D-A // check if at the last pixel

@START

D;JEQ // reset if at final pixel

@LOOP

D;JLT // keep painting black

@START

0;JMP // infinite loop to reset just in case

Fill.tstA screenshot of a cell phone

Description automatically generatedFillAutomatic.tstA screenshot of a social media post

Description automatically generated