Create all functions to be used

Main()

String vector

Looped to accommodate multiple inputs on same line

Loop through vectore and uppercase every input

If statement to see if its hex input

If so convert to binary to check which function it will get passed to

Other wise check if its assembly input

If so go to allocated function

Otherwise return back to main until correct input is found

Sbintohex()

Hextobin()

Dectobin()

Bintodec ()

^^^^^^^^

these converters were taken from blackboard since I couldn’t figure out how to do some of them.

Decodeadd(

If binary == placements 5,6&7 is value

Include add as assembler with srcreg and value

Else

Include add as assembler with srcgreg and destreg

)

Decodemoveloadstore()

If

Variable Binary== address

store

else if binary=placements 2,3 &4 is value

load

else

move{go to decodeadd for same result}

decodejump()

if hex first 2 input==80

jmp

else if first 2 input ==81

jpz

else if first 2 hex input == 82

jpp

else if first 2 hex input ==83

jpn

encodeaddandmove()

move/add=string

group of if statements to work out what the destreg and srcreg would be

otherwise reference the input allocated to the sbintodec for the number

encodeload()

load=string

loop an array to find out what the src reg converts to in binary

convert binary to hex

swap around the last input to create little endien

encodestore()

store=string

loop through array to find what the dest address is

convert binary to hex

swap 2nd inputted value to create little endien

encodejump()

group of if statements to determine what the 8bit binary would be

anything greater would be the little endien