

Leyla Mizrahi
om2243
Project 3 - Image Convolution

I worked on the project alone.

My design consists of one sub function that is called pixel which takes in the the address of the position of the byte to process, the kernel, and N which is how many pixels the image is.

If the kernel is

a b c

d e f

g h i

and the location adress x; the function will do

a * (x - 3N - 3)

b * (x - 3N)

c * (x - 3N + 3)

d * (x-3)

e * (x)

f * (x + 3)

g * (x + 3N - 3)

h * (x - 3N)

i * (x - 3N + 3)

I first put all my variables in to s registers. I than have a function advance that keeps branching to itself until N * 3 bytes are passed. I use a register as counter to decrement every time I advance the position. I branch to start once the counter is 0. I than move 3 more bytes.

I have two counters that are at N - 2 to keep the rows and columns to go.

I load N-2 into the location of a2 to set the size of the output.

I go into function row and check if my column counter is 0 if so brach to next row. call pixel on each byte of s0 which is a pointer to the image . I move s0 3 bytes brach back to row.

Next row will check if row counter is 0 brach to end if so. Increment the position by 6 bytes to get the start of the next line. Decrement the row counter, and set the column counter back to N-2.

End ends the conv function and returns back to main.