Problem EX-3 (1 part)  

MIPS and C

Part A: Consider the following MIPS code. Write a C program corresponding to this MIPS code by filling in body of the C program fragment provided below. Assume that the registers $2, $3, and $6 hold variables “Pixels,” “P,” and “Result,” respectively.

# Computes the OR of four 8-bit pixels packed in a 32-bit data word.
# $2: Pixels: packed data word
# $3: P: a single pixel
# $6: Result: bit-wise OR of all pixels
.data
Pixels: .word -12869361  # decimal equivalent of 0xFF3BA10F

.text
Dilate: lw $2, Pixels($0)
    andi $6, $2, 255  # mask: extract P0 (0xFF)
    srl $2, $2, 8    # logical shift right by 8 bits
    andi $3, $2, 255 # mask; extract P1
    srl $2, $2, 8    # shift off P1
    or $6, $3, $6    # $6: P1 or P0
    andi $3, $2, 255 # mask; extract P2
    srl $2, $2, 8    # shift off P2, $2 is right with just P3
    or $6, $3, $6    # $6: P2 or P1 or P0
    or $6, $2, $6    # $6: P3 or P2 or P1 or P0
jr $31

int Pixels = -12869361;
int main()
{
    int P;
    int Result= Pixels & 255;
    Pixels = Pixels >> 8;
    P = Pixels & 255;
    Result = P | Result;
    Pixels = Pixels >> 8;
    P = Pixels & 255;
    Result = P | Result;
    Pixels = Pixels >> 8;
    Result = Pixels | Result;
    return Result;
}