# decode2.c:

|  |
| --- |
| // decode2.c  // Homework Problem 3.58 decode2  #include <stdio.h>  //decode2(x,y,z)    //now lets get t1 = y – z.  //then cnompute t2 = x \* t1.  //C=and then compute t3 = (t1 << 63) >> 63,  //which is 0 if t1 ≥ 0, or −1 if t1 < 0.  //Finally, compute t4 = t2 XOR t3.  // This function returns t2 XOR t3, i.e. conditionally complement t2 when t1 is negative.    long decode2(long x, long y, long z) {      long t1 = y - z;                 // t1 = y – z      long t2 = x \* t1;                // t2 = x \* t1                                      // t2 = x \* (y – z) = x \* y – x \* z      long t3 = (t1 << 63) >> 63;      // t3 = sign‐mask of t1 (0 or −1)      long t4 = t3 ^ t2;               // t4 = t2 XOR t3      return t4;  }  // Main function to test decode2  int main(void) {      // Example usage of decode2      long x = 5, y = 2, z = 7;      long result = decode2(x, y, z);      printf("decode2(%ld, %ld, %ld) = %ld\n", x, y, z, result);      return 0;  } |

# decode2.s:

|  |
| --- |
| .file "decode2.c"  .text  .globl decode2  .type decode2, @function  decode2:  .LFB23:  .cfi\_startproc  endbr64  # Parameter-to-register mapping:  # x: %rdi  # y: %rsi  # z: %rdx  # Return value in %rax  subq %rdx, %rsi # Compute t1 = y - z  imulq %rsi, %rdi # Compute t2 = x \* t1  salq $63, %rsi # Shift t1 left by 63 to isolate sign bit  sarq $63, %rsi # Arithmetic right shift by 63 to get t3 = 0 or -1  movq %rdi, %rax # Move t2 to %rax  xorq %rsi, %rax # Compute t4 = t2 XOR t3  ret  .cfi\_endproc  .LFE23:  .size decode2, .-decode2  .section .rodata.str1.1,"aMS",@progbits,1  .LC0:  .string "decode2(%ld, %ld, %ld) = %ld\n"  .text  .globl main  .type main, @function  main:  .LFB24:  .cfi\_startproc  endbr64  subq $8, %rsp  .cfi\_def\_cfa\_offset 16  movl $7, %edx  movl $2, %esi  movl $5, %edi  call decode2  movq %rax, %r9  movl $7, %r8d  movl $2, %ecx  movl $5, %edx  leaq .LC0(%rip), %rsi  movl $1, %edi  movl $0, %eax  call \_\_printf\_chk@PLT  movl $0, %eax  addq $8, %rsp  .cfi\_def\_cfa\_offset 8  ret  .cfi\_endproc  .LFE24:  .size main, .-main  .ident "GCC: (Ubuntu 11.4.0-1ubuntu1~22.04) 11.4.0"  .section .note.GNU-stack,"",@progbits  .section .note.gnu.property,"a"  .align 8  .long 1f - 0f  .long 4f - 1f  .long 5  0:  .string "GNU"  1:  .align 8  .long 0xc0000002  .long 3f - 2f  2:  .long 0x3  3:  .align 8  4: |