CMPE 310 - Lab 1

The Files

- compile.sh

	Category	What it does
compile.sh	Helper File	This file runs a few commands that compiles and links assembly.
hammingDistance	Code File	This is the compiled exe of the code.
hammingDistance.asm	Code File	This is the assembly code of the program.
hammingDistance.o	Code File	
psudoCode.ts	Helper File	This file is TypeScript code for me to come up with a algorithm.
psudeoCode.txt	Helper File	This is the file that I used to translate the algorithm in `psudoCode.ts` to assembly
README.md	Git File	Just a readme
report.pdf	Lab Report	This is the final lab report file.

The Pseudo Code

```
BEGIN
    DISPLAY "First string: "
    READ str1
    NULL_TERMINATE(str1)

    DISPLAY "Second string: "
    READ str2
    NULL_TERMINATE(str2)

    SET hamming_distance = 0
    SET index = 0

    WHILE str1[index] is not null AND str2[index] is not null DO
        SET char1 = str1[index]
        SET char2 = str2[index]

        SET bit_position = 8
        WHILE bit_position > 0 DO
            IF (char1 LSB != char2 LSB) THEN
                INCREMENT hamming_distance
            END IF
            SHIFT_RIGHT char1
            SHIFT_RIGHT char2
            DECREMENT bit_position
        END WHILE

        INCREMENT index
    END WHILE

    CONVERT hamming_distance TO STRING hamming_distance_str

    DISPLAY "The Hamming distance is: "
    DISPLAY hamming_distance_str
```

```
    DISPLAY NEWLINE

    EXIT PROGRAM
END
```

The Code

```
section .data
    prompt_1 db "First string: ", 0
    prompt_2 db "Second string: ", 0
    result db "The Hamming distance is: ", 0
    newline db 10, 0
    hamming_distance_str db "000", 0

section .bss
    str1 resb 256
    str2 resb 256
    hamming_distance resb 1

section .text
    global _start

_start:
    ; first prompt
    mov eax, 4
    mov ebx, 1
    mov ecx, prompt_1
    mov edx, 14
    int 0x80

    ; read in string
    mov eax, 3
    mov ebx, 0
    mov ecx, str1
    mov edx, 255
    int 0x80

    ; null terminate
    mov byte [str1 + eax - 1], 0

    ; second prompt
    mov eax, 4
    mov ebx, 1
    mov ecx, prompt_2
    mov edx, 15
    int 0x80

    ; read in string
    mov eax, 3
    mov ebx, 0
    mov ecx, str2
    mov edx, 255
    int 0x80

    ; null terminate
    mov byte [str2 + eax - 1], 0

    ; calculate hamming distance
    xor ecx, ecx
    xor edx, edx

calculate_loop:
    mov al, [str1 + ecx]
    mov bl, [str2 + ecx]
    cmp al, 0
    je end_calculation
    cmp bl, 0
    je end_calculation

    ; Compare the binary representation of each character
    mov esi, 8

compare_bits:
    xor ah, ah
    xor bh, bh
    test al, 1
    setnz ah
```

```
    test bl, 1
    setnz bh
    cmp ah, bh
    je skip_bit
    inc edx

skip_bit:
    shr al, 1
    shr bl, 1
    dec esi
    jnz compare_bits

    inc ecx
    jmp calculate_loop

end_calculation:
    ; store data
    mov [hamming_distance], dl

    ; convert hamming distance to string
    movzx eax, byte [hamming_distance]
    call int_to_str

    ; print result string
    mov eax, 4
    mov ebx, 1
    mov ecx, result
    mov edx, 25
    int 0x80

    ; print result number (the hamming distance)
    mov eax, 4
    mov ebx, 1
    mov ecx, hamming_distance_str
    mov edx, 3
    int 0x80

    ; print the newline
    mov eax, 4
    mov ebx, 1
    mov ecx, newline
    mov edx, 1
    int 0x80

    ; close program
    mov eax, 1
    xor ebx, ebx
    int 0x80

int_to_str:
    ; Convert integer in EAX to string in hamming_distance_str
    mov ecx, 10
    xor ebx, ebx
    mov edi, hamming_distance_str + 2

convert_loop:
    xor edx, edx
    div ecx
    add dl, '0'
    mov [edi], dl
    dec edi
    test eax, eax
    jnz convert_loop
    ret
```

The Output

```
nick@nick-hpenvylaptop14teb000:~/School/cmpe310/HammingDistance$ ./hammingDistance
First string: foo
Second string: bar
The Hamming distance is: 008
nick@nick-hpenvylaptop14teb000:~/School/cmpe310/HammingDistance$ ./hammingDistance
First string: 560000000000124
Second string: 169807802340342
The Hamming distance is: 017
nick@nick-hpenvylaptop14teb000:~/School/cmpe310/HammingDistance$
```

Output of the code