CEN 214 Microprocessors Lab
Assignment 1

New Instruction Set Commands:

MOV [destination operand], [source operand]

Description: "Copies the second operand (source operand) to the first operand (destination operand). The source operand can be an immediate value, general-purpose register, segment register, or memory location; the destination register can be a general-purpose register, segment register, or memory location. Both operands must be the same size, which can be a byte, a word, or a doubleword[1]."

ADD [destination operand], [source operand]

Description: "Adds the first operand (destination operand) and the second operand (source operand) and stores the result in the destination operand. The destination operand can be a register or a memory location; the source operand can be an immediate, a register, or a memory location. (However, two memory operands cannot be used in one instruction.) When an immediate value is used as an operand, it is sign-extended to the length of the destination operand format.

The ADD instruction performs integer addition. It evaluates the result for both signed and unsigned integer operands and sets the OF and CF flags to indicate a carry (overflow) in the signed or unsigned result, respectively. The SF flag indicates the sign of the signed result. This instruction can be used with a LOCK prefix to allow the instruction to be executed atomically[2]."

SUB [destination operand], [source operand]

Description: "Subtracts the second operand (source operand) from the first operand (destination operand) and stores the result in the destination operand. The destination operand can be a register or a memory location; the source operand can be an immediate, register, or memory location. (However, two memory operands cannot be used in one instruction.) When an immediate value is used as an operand, it is sign-extended to the length of the destination operand format.

The SUB instruction performs integer subtraction. It evaluates the result for both signed and unsigned integer operands and sets the OF and CF flags to indicate an overflow in the signed or unsigned result, respectively. The SF flag indicates the sign of the signed result. This instruction can be used with a LOCK prefix to allow the instruction to be executed atomically[3]."

1 IA-32 Intel® Architecture Software Developer's Manual Volume 2: Instruction Set Reference, 2003, (MOV Command, pp. 3-441)
2 ...(ADD Command, pp. 3-22)
3 ...(SUB Command, pp. 3-756)

Examples:

1. Write an Intel x86 assembly code that will write **34h** value to **0100:1000h** and **0100:2000h** memory addresses.

2. Write an Intel x86 assembly code that will swap the values of **CL** and **DL** registers.

3. Write an Intel x86 assembly code that will add **98h** value written on **0100:0500h** memory address with **52h** value written on **0100:0501h** memory address, than write the result to **0100:0502h** memory address.