

Q1: Write a program to multiply two numbers of one byte. Use variable to store one number which is 255 and other value is also 255.

Note:

- I. One byte cannot hold a value greater than 255 so, the product cannot exceed two bytes i.e. 65, 025.
- II. When two bytes are multiplied, the multiplicand is in the AL register, and the multiplier is a byte in the memory or in another register.
- III. The product is in AX. High-order 8 bits of the product is stored in AH and the low-order 8 bits are stored in AL.

Q2: Write a program to multiply two numbers of one word. One number is 255 and other is 256. Store 256 in AX register.

Experiment the below mentioned scenarios.

- I. Store 255 in a variable of byte type.
- II. Store 255 in a variable of word type.

Note:

- I. When two one-word values are multiplied, the multiplicand should be in AX register, and the multiplier is a word in memory or register.
- II. High-order 16 bits of the product is stored in DX and the low-order 16 bits are stored in AX.
- III. If AH has a value, then the multiplicand is of one word. Here only two-word multiplication is applicable.
- IV. Suppose multiplier is in variable A of word type then multiplicand should be stored in AX.

Q3: Write a program to divide two numbers. The dividend is 65535 and divisor is 2.

- I. Store the dividend in 16bit register and perform a division in which the divisor is of 1 byte.
- II. Store the dividend in 16bit register and initialize DX to zero. Perform a division in which the divisor is of 1 word.

Note:

- I. When divisor is of 1 byte, the dividend is assumed to be in the AX register (16 bits). After division, the quotient goes to the AL register and the remainder goes to the AH register.
- II. While performing division you should know the maximum possible outcome. In case of 65535 the maximum quotient can be achieved by dividing by a smallest value i.e. 2.
- III. Since the value of quotient exceeds 8 bits, so for division the use of 1-word divisor is necessary, even the divisor is smaller than 1 word.
- IV. When divisor is of 1 word, the dividend is assumed to be 32 bits long and in the DX:AX registers. The high-order 16 bits are in DX and the low-order 16 bits are in AX. After division, the 16-bit quotient goes to the AX register and the 16-bit remainder goes to the DX register.

Q4: Write a program to read a one-digit divisor from the user. Divide the number 25 by the divisor.

- I. Store 25 in AL, without changing AH.**
 - II. Store 25 in AX**
- or**
- Store 25 in AL but initialize AH to 0.**

Note:

- I. After taking input, convert the number to its actual decimal value.
- II. Before division initialize AH to zero to avoid division of a different value as 25 will be in AL and input function code 01 will change the overall value in AX register.
- III. When divisor is of 1 word, sometimes initializing DX to 0 is also required.

Q5: Make three variables (v1, v2, v3). Read two values from the user and store them in v1 and v2, respectively. Add first two values in variables. Take third value from the user and store it in v3. Suppose the 3rd value is of one digit and smaller than the sum of two previous values. Subtract the value in third variable from the result obtained after the addition of v1 and v2. Display the output on the screen.