**Задание 3.1**. В регистре AX задано число от 0 до 65535. Выведите это число на экран. (Проверить программу для числа более 255.)

.model tiny  
.386  
.code  
org 100h  
start:  
  
call ShowText  
mov AX, 65535  
mov BX, 10  
mov CX, 0  
label1:  
mov DX, 0  
div BX  
push DX  
add CX, 1  
cmp AX, 0  
jnz label1  
label2:  
pop DX  
call ShowNumber  
loop label2  
mov AX, 4C00h  
int 21h  
  
ShowNumber proc  
push AX  
mov AH, 02h  
add DL, 30h  
int 21h  
pop AX  
ret  
ShowNumber endp  
  
ShowText proc  
mov AH, 09h  
mov DX, offset Text  
int 21h  
ret  
ShowText endp  
  
Text db 'Kuzichkin Pavel, 241st group', 0Ah, '$'  
  
end start



**Задание 3.2**. Используя 32-битные регистры процессора, напишите программу, выводящую на экран число 65536.

model tiny  
.386  
.code  
org 100h  
start:  
  
call ShowText  
  
mov EAX, 65536  
mov BX, 10  
mov CX, 0  
label1:  
mov EDX, 0  
div EBX  
push EDX  
inc CX  
cmp EAX, 0  
jnz label1  
label2:  
pop EDX  
call ShowNumber  
loop label2  
  
mov EAX, 4C00h  
int 21h  
  
ShowNumber proc  
push EAX  
mov AH, 02h  
add DL, 30h  
int 21h  
pop EAX  
ret  
ShowNumber endp  
  
ShowText proc  
mov AH, 09h  
mov DX, offset Text  
int 21h  
ret  
ShowText endp  
  
Text db 'Kuzichkin Pavel, 241st group', 0Ah, '$'  
  
end start



**Контрольные вопросы и упражнения**

**1)** Чем отличается деление на байт от деления на слово? (где должно располагаться делимое, куда попадут частное от деления и остаток от деления)

**Ответ:** Байтовая команда делит 16-битовое делимое на 8-битовый делитель. Делимое находится в регистре AX. В результате деления получается два числа: частное помещается в регистр AL, а остаток – в AH.

Команда, работающая со словами, делит 32-битовое делимое на 16-битовый делитель. Делимое находится в паре регистров DX: AX, причем регистр DX содержит старшую значимую часть, а регистр AX – младшую. Команда деления помещает частное в регистр AX, а остаток в DX.

**2)** Как формируется машинный код для команд безусловного перехода типа SHORT, NEAR и FAR?

**Ответ:** SHORT: на расстоянии -128 бит до 128 бит (2 байт); NEAR: до 32678 бит (3 байта); FAR: в другом кодовом сегменте (5 байт)

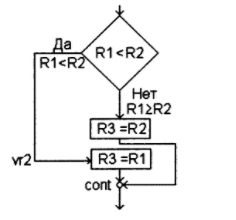
**3)** Каков механизм действия команды cmp? В паре с какими командами она обычно используется?

**Ответ:** Команда сравнения CMP сравнивает два числа, вычитая второе из первого. Инструкция CMP не сохраняет результат, а лишь устанавливает в соответствии с результатом флаги состояния. *Основное назначение команды CMP – это организация ветвлений (условных переходов) в ассемблерных программах*.

**4)** На какие флаги реагируют команды условного перехода для чисел со знаком и для чисел без знака?

**Ответ:** OF, CF, ZF, PF, SF

**5)** С помощью команд условного и безусловного перехода выполните программную реализацию алгоритма ветвления для определения наименьшего числа из двух заданных. Алгоритм изображен в виде блок-схемы, приведенной на рис.



**Ответ:** cmp ax, bx ;сравнение ax,bx

jl vr2 ;если ax < bx переходим в vr2

mov dx,bx

jmp cont ;иначе переходим в cont

vr2: mov dx, ax

cont: … выход

**6)** Каков механизм работы команды организации цикла LOOP?

**Ответ:** Она в конце каждой итерации уменьшает содержимое CX на 1 и передает управление на метку (указанную в команде), если содержимое CX не равно 0. Если вычитание 1 из CX привело к нулевому результату, выполняется следующая команда.

**7)** Как с помощью команды сдвига можно поделить знаковое число, хранящееся в АХ, на 2 в n-ой степени?

**Ответ:** Mov Ax,число

MOV CL, n

SHR AX,CL

0000 0000 0000 0111

**8)** Как с помощью команды сдвига проверить содержимое регистра ВХ на четность?

**Ответ:** shr BX,1 => if CF=1 = not even, else even.