

# Основы языка ассемблер

# Операции ввода/вывода

# Операции ввода/вывода по прерыванию INT 21H

- ▶ **АН = 01: Ввод с клавиатуры с эхоотображением.** Данная функция возвращает значение в регистре AL.
- ▶ **АН = 02: Вывод символа.** Для ввода символа на экран в текущую позицию курсора необходимо поместить код данного символа в регистр DL.
- ▶ **АН = 06: Ввод/вывод данных.** Может использоваться как для ввода, так и для вывода. Для вывода занесите в DL выводимый символ (но не FFH!) и прерывание 21H. Для ввода в DL занесите FFH, выполните прерывание 21H. Программа при этом не останавливается, продолжает выполняться. При нажатии клавиши символ вводится в AL.
- ▶ **АН = 07: Прямой ввод с клавиатуры без эхоотображения.** Данная функция работает аналогично функции 01.
- ▶ **АН = 08: Ввод с клавиатуры без эхоотображения.** Данная функция действует аналогично функции 01 с одним отличием: введенный символ не отображается на экране, т.е. нет эха.
- ▶ **АН = 09: Вывод строки символов.** Выводимая строка должна заканчиваться знаком доллара \$. Адрес начала строки должен быть помещен в DX.
- ▶ **АН=0AH: Ввод данных в буфер:** Определяется максимальная длина вводимого текста. Это необходимо для предупреждения пользователя звуковым сигналом, если набран слишком длинный текст; символы, превышающие максимальную длину, не принимаются. Во второй байт буфера команда возвращает действительную длину введенного текста в байтах. Адрес буфера помещается в регистр DX.
- ▶ **АН = 0BH: Проверка состояния клавиатуры.** Данная функция возвращает шестнадцатеричное значение FF в регистре AL, если ввод с клавиатуры возможен, в противном случае - 00.

# Работа со стеком

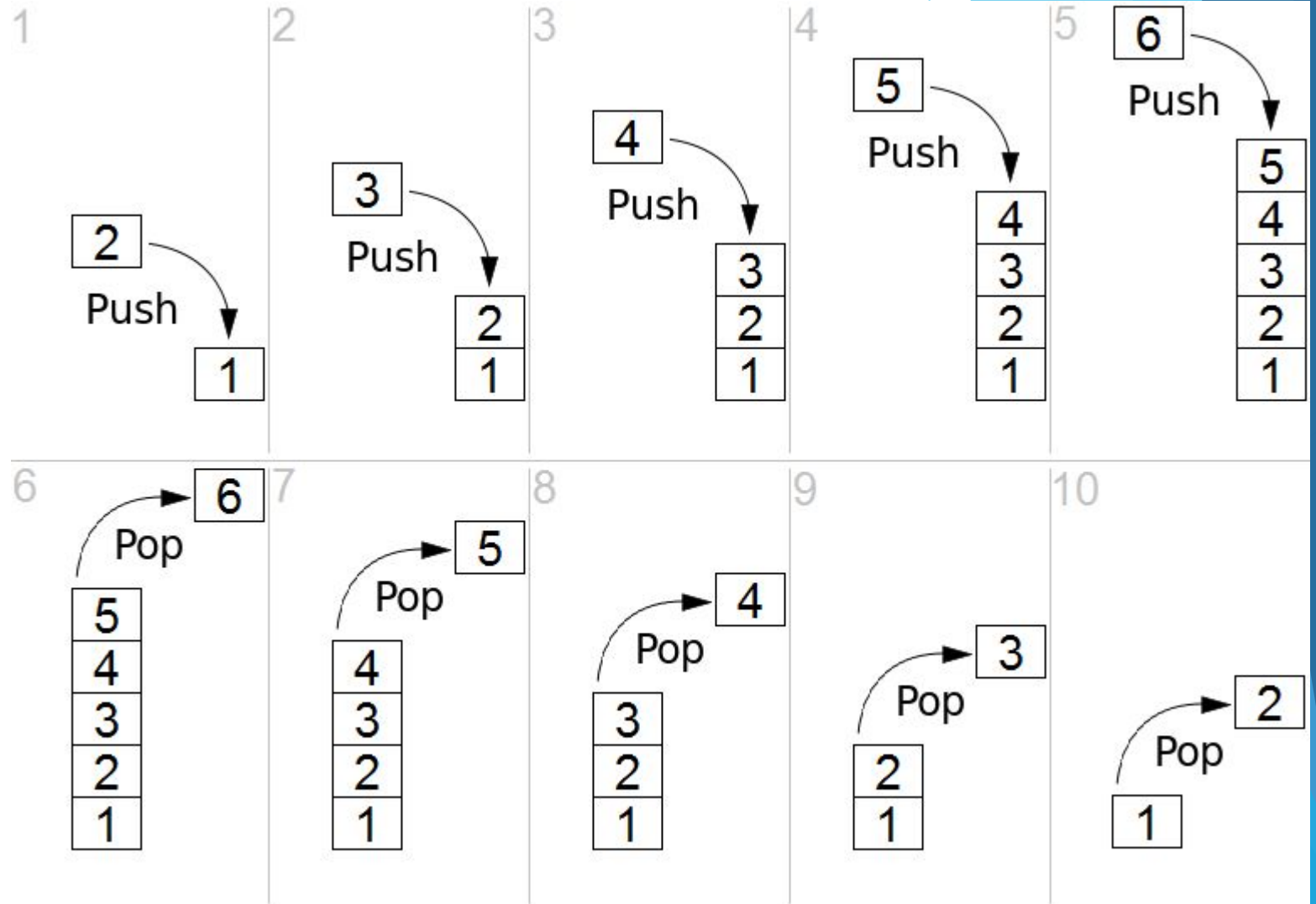
**push dx**

;поместить значение ;  
регистра dx в стек

**pop dx**

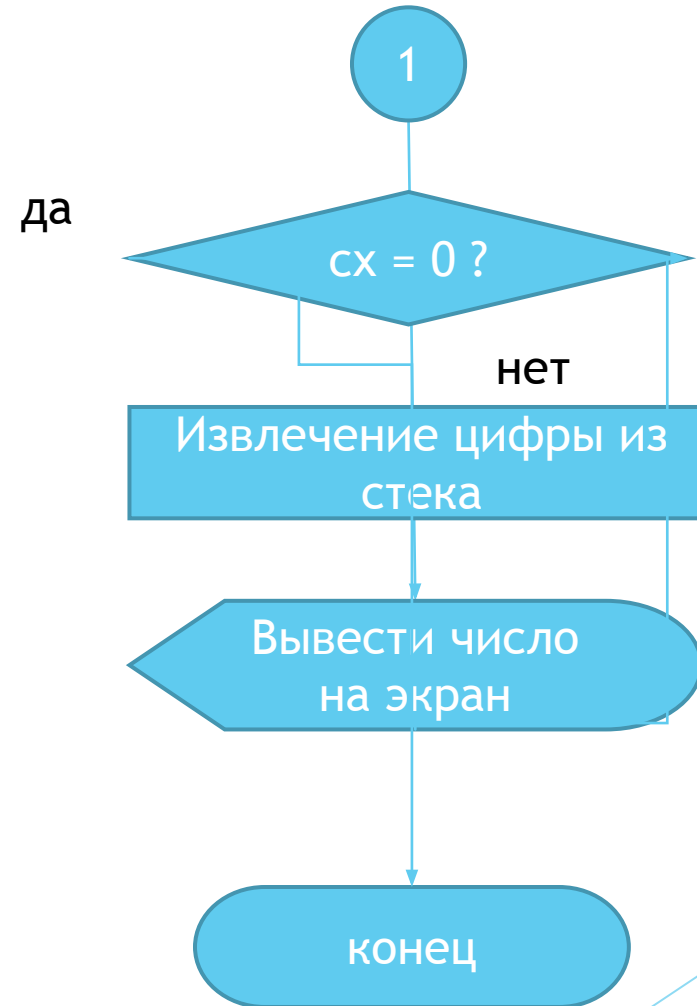
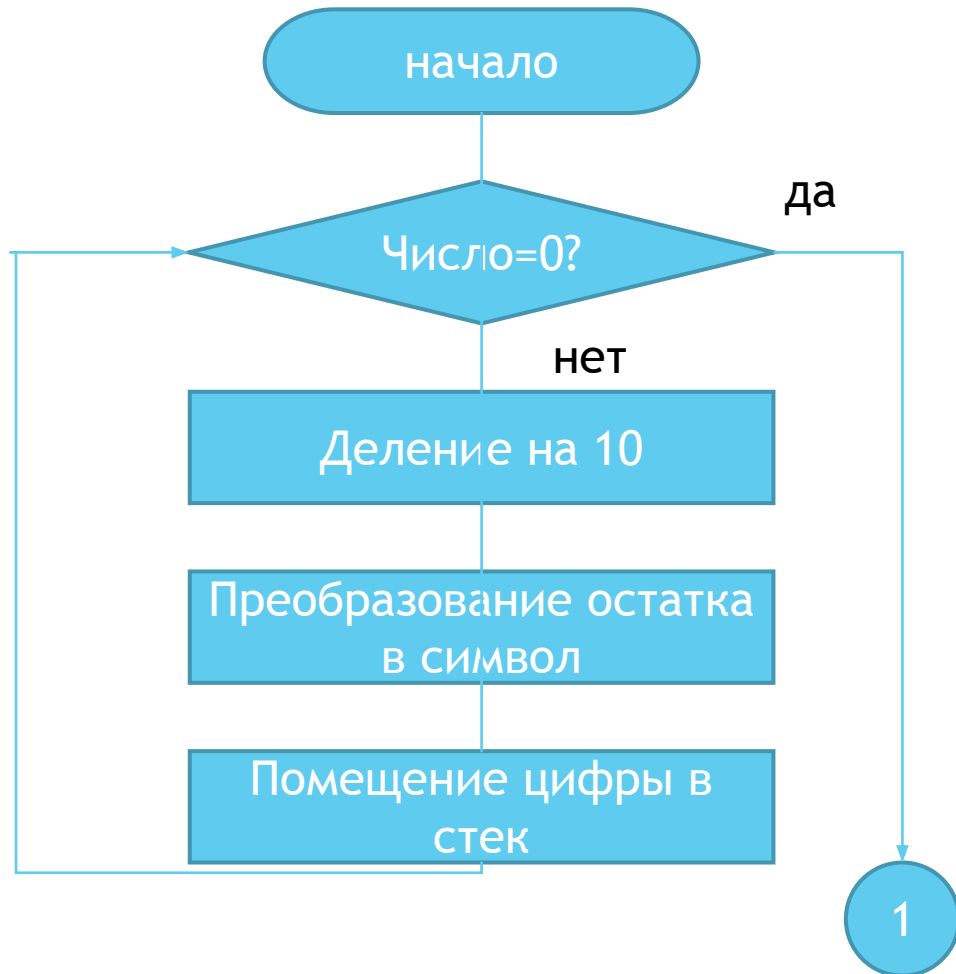
;записать в dx число

;из стека



# Задача

Вывести значение переменной на экран.



# Программа

```
org 100h
.model tiny
chislo dd 12345      ;число для вывода
.code
convert:
mov ax, chislo      ;число в AX
mov cx,0            ; CX=0 – счетчик цифр
mov bx,10           ; BX=0 – основание системы
                    ; счисления

.divide:
mov dx,0            ;DX = 0;
div bx              ;деление AX на BX,
                    ;частное в AX
                    ;остаток в DX
add dx, '0'         ;добавление ASCII-код
                    ; цифры 0 к остатку
push dx             ;сохранение в стеке
inc cx              ;счетчик +1

cmp ax,0            ;все? (Частное = 0?)
jnz .divide         ;если нет
```

# Задача

1. Найти сумму двух чисел введенных с клавиатуры, если сумма  $< 10$ .
2. Найти сумму двух чисел введенных с клавиатуры для любой суммы.

# Программа

```
.model tiny
org 100h
.data
    stroka1 DB 'Введите первую цифру ','$'
    stroka2 DB 13,10,'Введите вторую цифру ','$'
    stroka3 DB 13,10,'Сумма равна  $'

.code
    mov  ah,9           ; функция вывода строки
    mov  dx,offset stroka1 ; ссылка на stroka1
    int  21h           ; вывести stroka1 на экран
    mov  ah,1          ; функция ввода символа
    int  21h           ; ввод 1-го символа
    sub  al,30h        ; преобразование 1-го
                       ; символа в цифру
    push ax            ; перемещение в стек
    mov  ah,9          ; функция вывода строки
    mov  dx,offset stroka2 ; ссылка на stroka2
    int  21h           ; вывести stroka2 на экран
    mov  ah,1          ; функция ввода символа
    int  21h           ; ввод 2-го символа
    sub  al,30h        ; преобразование 2-го символа
                       ; в цифру
```



# ВВОД С КЛАВИАТУРЫ ПО КОМАНДЕ BIOS INT 16H

Команда BIOS INT 16H выполняет специальную операцию, которая в соответствии с кодом в регистре AH обеспечивает следующие три функции ввода с клавиатуры.

- ▶ **AH = 00: Чтение символа.** Данная функция помещает в регистр AL очередной ASCII-символ, введенный с клавиатуры, и устанавливает скэн-код в регистре AH. Если на клавиатуре нажата одна из специальных клавиш, например, Home или F1, то в регистр AL заносится 00. Автоматическое эхо символа на экран не происходит.
- ▶ **AH = 01: Определение наличия введенного символа.** Данная функция сбрасывает флаг нуля (ZF=0), если имеется символ для чтения с клавиатуры; очередной символ и скэн-код будут помещены в регистры AL и AH соответственно и данный элемент останется в буфере.
- ▶ **AH = 02: Определение текущего состояния клавиатуры.** Данная функция возвращает в регистре AL состояние клавиатуры из адреса памяти 417H:\

7	Состояние вставки активно (Ins)	3	Нажата комбинация клавиш Alt/Shft
6	Состояние фиксации верхнего регистра (Caps Lock) включено	2	Нажата комбинация клавиш Ctrl/Shft
5	Состояние фиксации цифровой клавиатуры (Num Lock) включено	1	Нажата левая клавиша Shift
4	Состояние фиксации прокрутки (Scroll Lock) включено	0	Нажата правая клавиша Shift

# Задания

- В цикле ввести символ с клавиатуры и вывести его двоичное представление на экран. Если введен символ \*, закончить работу программы.
- В цикле ввести десятичное число с клавиатуры (Функция `AH=2 INT 21H`). Число десятичных разрядов от 1 до 5. Признак конца ввода - нажатие клавиши [Ввод] (код 13). Преобразовать число в двоичное и вывести его двоичное представление на экран.
- В цикле ввести десятичное число с клавиатуры (Функция `AH=0AH INT 21H`).. Число десятичных разрядов от 1 до 5. Признак конца ввода - нажатие клавиши [Ввод] (код 13). Преобразовать число в двоичное и вывести его двоичное представление на экран.
- Введите 8 битов с клавиатуры - последовательность 0 и 1. Выведите на экран преобразованную последовательность в виде символа ASCII в заданной позиции экрана, которая вводится с клавиатуры (строка, столбец).