

.Model small

.386

.data

```
string db '(-: Laboratornaya SPZ№8. Ver2.03 Copyright(c) 06.05.2003. Olexander Bilovol :-),'$'  
;  
; Правильно работает изменение системной даты, присвоена версия 2.00;  
; установлены временные ограничения функциональности программы, начаты попытки  
; оптимизации кода.  
;string db '(-: Laboratornaya SPZ№8. Ver1.11 Copyright(c) 06.05.2003. Olexander Bilovol :-),'$'  
;  
; Правильно выводится дата и время, присвоена версия 1.00  
;string db '(-: Laboratornaya SPZ№8. Ver0.16 Copyright(c) 06.05.2003. Olexander Bilovol :-),'$'  
date db 'Дата : ','$' ; Объявление переменных, которые хранят строки  
time db 'Время : ','$' ; для вывода на экран.  
date_string db 'ДД-ММ-ГГ, день недели','$' ; Строка подсказки даты  
time_string db 'ЧЧ:ММ.сс','$' ; Строка подсказки времени
```

```
date_weack_0 db ', Воскресенье','$' ; День недели  
date_weack_1 db ', Понедельник','$' ; День недели  
date_weack_2 db ', Вторник','$' ; День недели  
date_weack_3 db ', Среда','$' ; День недели  
date_weack_4 db ', Четверг','$' ; День недели  
date_weack_5 db ', Пятница','$' ; День недели  
date_weack_6 db ', Суббота','$' ; День недели
```

```
;month_1 db 'Январь','$' ;  
;month_2 db 'Февраль','$' ;  
;month_3 db 'Март','$' ;  
;month_4 db 'Апрель','$' ;  
;month_5 db 'Май','$' ;  
;month_6 db 'Июнь','$' ;  
;month_7 db 'Июль','$' ;  
;month_8 db 'Август','$' ;  
;month_9 db 'Сентябрь','$' ;  
;month_a db 'Октябрь','$' ;  
;month_b db 'Ноябрь','$' ;  
;month_c db 'Декабрь','$' ;
```

```
month_1 db '01','$' ; Январь  
month_2 db '02','$' ; Февраль  
month_3 db '03','$' ; Март  
month_4 db '04','$' ; Апрель  
month_5 db '05','$' ; Май  
month_6 db '06','$' ; Июнь  
month_7 db '07','$' ; Июль  
month_8 db '08','$' ; Август  
month_9 db '09','$' ; Сентябрь  
month_a db '10','$' ; Октябрь  
month_b db '11','$' ; Ноябрь  
month_c db '12','$' ; Декабрь
```

```
Zapros_i_can_do db 'Для изменения нажмите один из вариантов:','$'  
Zapros_Data db 'Изменить дату - 1.','$'  
Zapros_Time db 'Изменить время - 2.','$'
```

```
Zapros_refersh      db 'Обновить показания времени - 3.','$'
Zapros_exit  db 'Выйти из программы      - 0.','$'
```

```
New_hour db ?
new_minute db ?
New_day db ?
New_month db ?
New_Year dw ?
New_Year_to_write db ?
```

```
;!!!!!!!!!!!!!!!!!!!!!!
;
;      ОГРАНИЧЕНИЕ, УСТАНОВЛЕННОЕ В ПРОГРАММЕ:
;
; 1.   Для значительного упрощения программы было решено временно ограничить
; корректный диапазон корректного вывода значения текущего года (номера)
; диапазоном: с 2000 по 2003 годы и не изменяется текущий год с выводом при изменении
2003.
;
; 2.   Для упрощения программы было решено временно отключить ввод нового
; значения текущего года (номера) и устанавливается значение текущего.
;
; 3.   Для упрощения программы было решено временно отложить доработку
; и изменение значения времени.
;
;
;      ОГРАНИЧЕНИЕ, УСТАНОВЛЕННОЕ В ПРОГРАММЕ.
;
;!!!!!!!!!!!!!!!!!!!!!!

;      Сообщение об ошибке :
error db ' !!! ERROR - ОШИБКА - ERROR !!! ','$'
```

```
.code
```

```
Ochistka_Ecrana Proc
```

```
; Очистка экрана:
;-----
;Подготовка до очищения - встановлення змін, котрі не можуть змінюватися.
mov cx,25      ;Кількість строк для очищення
mov dh,00h     ;Початкова строка для очищення.

cikl:
;Позиціювання курсору у лівий верхній кут монітору.
;-----
mov ah,02h     ;Код переривання.
mov bh,00h     ;Номер сторінки екрану.
```

```

        mov dl,00h    ;Стовпець.
        int 10h       ;Виклик переривання.
;=====

;Очищення екрану монітору шляхом запису строки з символів ' '.
;-----
        mov ah,0ah    ;Код переривання.
        mov al,' '    ;Символ для повторення.
        mov bx,0;Номер сторінки екрану (всього їх 8+2 графічні).
        push cx       ;Збереження стану регістру.
        mov cx,80     ;Кількість повторень символу(Курсор з позиції не зсувається).
        int 10h       ;Виклик переривання.
        pop cx
        inc dh
        loop cikl
;=====

;Позиціювання курсору у лівий верхній кут монітору.
;-----
        mov ah,02h    ;Код переривання.
        mov bh,00h    ;Номер сторінки екрану.
        mov dh,01h    ;Строка
        mov dl,01h    ;Стовпець.
        int 10h       ;Виклик переривання.
;=====
        ret
;=====
Ochistka_Ecrana Endp

```

Podscaska_Date Proc

```

        ;Формирование строки подсказки для даты:
;-----
        ;Вивід строки
;-----
        mov dx,offset date
        mov ah,09h    ;Код переривання.
        int 21h       ;Виклик переривання.
;=====

        ;Позиціювання курсору на наступну строку.
;-----
        mov ah,02h    ;Код переривання.
        mov bh,00h    ;Номер сторінки екрану.
        mov dh,02h    ;Строка
        mov dl,0ah    ;Стовпець.
        int 10h       ;Виклик переривання.
;=====
        ;Вивід строки
;-----
        mov dx,offset date_string
        mov ah,09h    ;Код переривання.
        int 21h       ;Виклик переривання.

```

```

;=====
ret
;=====
Podscaska_Date Endp

```

Podscaska_time Proc

```

;-----
;Позиціювання курсору на наступну строку.
;-----
mov ah,02h ;Код переривання.
mov bh,00h ;Номер сторінки екрану.
mov dh,04h ;Строка
mov dl,0ah ;Стовпець.
int 10h ;Виклик переривання.
;=====
;Вивід строки
;-----
mov dx,offset time_string
mov ah,09h ;Код переривання.
int 21h ;Виклик переривання.
;=====
ret
;=====
Podscaska_time Endp

```

Write_day_number Proc

```

;-----
; dl - Номер Дня месяца, который надо вывести.
;Позиціювання курсору.
;-----
mov ah,02h ;Код переривання.
mov bh,00h ;Номер сторінки екрану.
mov dh,01h ;Строка
mov dl,0bh ;Стовпець.
int 10h ;Виклик переривання.
;=====
;Получение значения текущей даты:
;-----
sub ax,ax ; Для подготовки проводится очистка регистра AX
mov ah,2ah ; Установка номера прерывания
int 21h ; Выполнение прерывания
;=====
; dl - Номер Дня месяца, который надо вывести.
cmp dl,30
jle date_30
;Вывод даты 31
;=====
sub dl,30 ; Вычесть из текущего дня 30
add dl,48 ; Дополнить полученное до ASCII - кода.
mov ah,02h ; Выбор функции прерывания.
int 21h ; Вызов прерывания.

```

```

;=====
;Позиціювання курсору.
;-----
    mov ah,02h    ;Код переривання.
    mov bh,00h    ;Номер сторінки екрану.
    mov dh,01h    ;Строка
    mov dl,0ah    ;Стовпець.
    int 10h       ;Виклик переривання.
;=====

    mov dl,51     ; Отправить в dl ASCII - код цифры 3.
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
    jmp end_date

date_30:         ;Вывод даты 30
    cmp dl,30
    jl date_20
    ;Позиціювання курсору.
;-----
    mov ah,02h    ;Код переривання.
    mov bh,00h    ;Номер сторінки екрану.
    mov dh,01h    ;Строка
    mov dl,0ah    ;Стовпець.
    int 10h       ;Виклик переривання.
;=====

    mov dl,51     ; Отправить в dl ASCII - код цифры 3.
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
    jmp end_date

date_20:         ;Вывод даты с 20 по 29
    cmp dl,20
    jl date_10
;=====

    sub dl,20     ; Вычесть из текущего дня 30
    add dl,48     ; Дополнить полученное до ASCII - кода.
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
;=====

    ;Позиціювання курсору.
;-----
    mov ah,02h    ;Код переривання.
    mov bh,00h    ;Номер сторінки екрану.
    mov dh,01h    ;Строка
    mov dl,0ah    ;Стовпець.
    int 10h       ;Виклик переривання.
;=====

    mov dl,50     ; Отправить в dl ASCII - код цифры 2.
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
    jmp end_date

```

```

date_10:      ;Вывод даты с 10 по 19
              cmp dl,10
              jl date_00
;=====
              sub dl,10      ; Вычесть из текущего дня 10
              add dl,48      ; Дополнить полученное до ASCII - кода.
              mov ah,02h     ; Выбор функции прерывания.
              int 21h        ; Вызов прерывания.
;=====
              ;Позиціювання курсору.
;-----
              mov ah,02h     ;Код переривання.
              mov bh,00h     ;Номер сторінки екрану.
              mov dh,01h     ;Строка
              mov dl,0ah     ;Стовпець.
              int 10h        ;Виклик переривання.
;=====
              mov dl,49      ; Отправить в dl ASCII - код цифры 1.
              mov ah,02h     ; Выбор функции прерывания.
              int 21h        ; Вызов прерывания.
              jmp end_date

date_00:      ;вывод даты с 00 по 9

              add dl,48      ; Дополнить полученное до ASCII - кода.
              mov ah,02h     ; Вызов прерывания.
              int 21h

end_date:
              ret
;=====
Write_day_number EndP

Write_day_of_weak proc
              ;Процедура вывода на экран текущего дня недели
;-----
;              al - День недели, который надо вывести.
              ;Позиціювання курсору на позицію.
;-----
              mov ah,02h     ;Код переривання.
              mov bh,00h     ;Номер сторінки екрану.
              mov dh,01h     ;Строка
              mov dl,12h     ;Стовпець.
              int 10h        ;Виклик переривання.
;=====

              ;Получение значения текущей даты:
;-----
              sub ax,ax      ; Для подготовки проводится очистка регистра AX
              mov ah,2ah     ; Установка номера прерывания
              int 21h        ; Выполнение прерывания

```

```

;=====

    cmp al,6
    jnz str_5

    ;Вивід строки
;-----
    mov dx,offset date_weack_6
    mov ah,09h    ;Код переривання.
    int 21h      ;Виклик переривання.
;=====

    jmp ex

str_5:
    cmp al,5
    jnz str_4

    ;Вивід строки
;-----
    mov dx,offset date_weack_5
    mov ah,09h    ;Код переривання.
    int 21h      ;Виклик переривання.
;=====

    jmp ex

str_4:
    cmp al,4
    jnz str_3

    ;Вивід строки
;-----
    mov dx,offset date_weack_4
    mov ah,09h    ;Код переривання.
    int 21h      ;Виклик переривання.
;=====

    jmp ex

str_3:
    cmp al,3
    jnz str_2

    ;Вивід строки
;-----
    mov dx,offset date_weack_3
    mov ah,09h    ;Код переривання.
    int 21h      ;Виклик переривання.
;=====

    jmp ex

str_2:
    cmp al,2
    jnz str_1

```

```

        ;Вивід строки
;-----
        mov dx,offset date_weack_2
        mov ah,09h      ;Код переривання.
        int 21h         ;Виклик переривання.
;=====

        jmp ex

str_1:
        cmp al,1
        jnz str_0

        ;Вивід строки
;-----
        mov dx,offset date_weack_1
        mov ah,09h      ;Код переривання.
        int 21h         ;Виклик переривання.
;=====

        jmp ex

str_0:
        ;Вивід строки
;-----
        mov dx,offset date_weack_0
        mov ah,09h      ;Код переривання.
        int 21h         ;Виклик переривання.
;=====

ex: ret
;=====
Write_day_of_weak Endp

Write_month_name proc
;-----
        ;Позиціювання курсору.
;-----
        mov ah,02h      ;Код переривання.
        mov bh,00h      ;Номер сторінки екрану.
        mov dl,0dh      ;Стовпець.
        mov dh,01h      ;Строка
        int 10h         ;Виклик переривання.
;=====

        ;Получение значения текущей даты:
;-----
        sub ax,ax        ; Для подготовки проводится очистка регистра AX
        mov ah,2ah       ; Установка номера прерывания
        int 21h          ; Выполнение прерывания
;=====
;        dh - Номер года, который надо вывести.
        mov ch,dh

```



```
    cmp ch,0ch
    jnz monthb
```

```
    ;Вивід строки
```

```
;------
    mov dx,offset month_c
    mov ah,09h      ;Код переривання.
    int 21h         ;Виклик переривання.
;=====
    jmp ex_month
```

```
monthb:
```

```
    cmp ch,0bh
    jnz montha
```

```
    ;Вивід строки
```

```
;------
    mov dx,offset month_b
    mov ah,09h      ;Код переривання.
    int 21h         ;Виклик переривання.
;=====
    jmp ex_month
```

```
montha:
```

```
    cmp ch,0ah
    jnz month9
```

```
    ;Вивід строки
```

```
;------
    mov dx,offset month_a
    mov ah,09h      ;Код переривання.
    int 21h         ;Виклик переривання.
;=====
    jmp ex_month
```

```
month9:
```

```
    cmp ch,9
    jnz month8
```

```
    ;Вивід строки
```

```
;------
    mov dx,offset month_9
    mov ah,09h      ;Код переривання.
    int 21h         ;Виклик переривання.
;=====
    jmp ex_month
```

```
month8:
```

```
    cmp ch,8
    jnz month7
```

```

        ;Вивід строки
;-----
        mov dx,offset month_8
        mov ah,09h      ;Код переривання.
        int 21h         ;Виклик переривання.
;=====
        jmp ex_month

month7:
        cmp ch,7
        jnz month6

        ;Вивід строки
;-----
        mov dx,offset month_7
        mov ah,09h      ;Код переривання.
        int 21h         ;Виклик переривання.
;=====
        jmp ex_month

month6:
        cmp ch,6
        jnz month5

        ;Вивід строки
;-----
        mov dx,offset month_6
        mov ah,09h      ;Код переривання.
        int 21h         ;Виклик переривання.
;=====
        jmp ex_month

month5:
        cmp ch,5
        jnz month4

        ;Вивід строки
;-----
        mov dx,offset month_5
        mov ah,09h      ;Код переривання.
        int 21h         ;Виклик переривання.
;=====
        jmp ex_month

month4:
        cmp ch,4
        jnz month3

        ;Вивід строки
;-----
        mov dx,offset month_4
        mov ah,09h      ;Код переривання.

```

```

        int 21h      ;Виклик переривання.
;=====

        jmp ex_month

month3:
        cmp ch,3
        jnz month2

        ;Вивід строки
;-----
        mov dx,offset month_3
        mov ah,09h    ;Код переривання.
        int 21h      ;Виклик переривання.
;=====
        jmp ex_month

month2:
        cmp ch,2
        jnz month1

        ;Вивід строки
;-----
        mov dx,offset month_2
        mov ah,09h    ;Код переривання.
        int 21h      ;Виклик переривання.
;=====
        jmp ex_month

month1:
        ;Вивід строки
;-----
        mov dx,offset month_1
        mov ah,09h    ;Код переривання.
        int 21h      ;Виклик переривання.
;=====
        jmp ex_month

ex_month:
        ret
;=====
Write_month_name Endp

Write_year_number Proc
;-----
;Вывод Номера текущего года в диапазоне с 2000 по 2009 годы
;Позиціювання курсору.
;-----
        mov ah,02h    ;Код переривання.
        mov bh,00h    ;Номер сторінки екрану.
        mov dl,11h    ;Стовпець.
        mov dh,01h    ;Строка
        int 10h      ;Виклик переривання.

```

```

;=====
;Получение значения текущей даты:
;-----
    sub ax,ax      ; Для подготовки проводится очистка регистра AX
    mov ah,2ah     ; Установка номера прерывания
    int 21h        ; Выполнение прерывания
°;=====

    mov New_year,cx ; Сохранение текущего года для повторной установки.

    sub cx,7d0h    ; Вычесть из текущего года 2000
    add cx,48      ; Дополнить полученное до ASCII - кода.
    mov dl,cl      ; Скопировать значение года в dl для вывода на экран.
    mov ah,02h     ; Вызов прерывания.
    int 21h

;=====
    ret
Write_year_number Endp

Writte_date Proc
;-----
;Получение значения текущей даты:
;-----
    sub ax,ax      ; Для подготовки проводится очистка регистра AX
    mov ah,2ah     ; Установка номера прерывания
    int 21h        ; Выполнение прерывания
;=====
;Вывод нулевого значения даты.
;-----
;Позиціювання курсору на позицію.
;-----
    mov ah,02h     ;Код переривання.
    mov bh,00h     ;Номер сторінки екрану.
    mov dh,01h     ;Строка
    mov dl,0ah     ;Стовпець.
    int 10h        ;Виклик переривання.
;=====
;Вывод восьми цифр 00
;-----
    mov ah,0ah
    mov al,48
    mov cx,08h
    int 10h
;=====
;Позиціювання курсору на позицію.
;-----
    mov ah,02h     ;Код переривання.
    mov bh,00h     ;Номер сторінки екрану.
    mov dh,01h     ;Строка
    mov dl,0ch     ;Стовпець.
    int 10h        ;Виклик переривання.
;=====

```

```

        ;Вывод первого символа "-"
;-----
        mov ah,0ah
        mov al,45
        mov cx,01h
        int 10h
;=====

        ;Позиціювання курсору на позицію.
;-----
        mov ah,02h    ;Код переривання.
        mov bh,00h    ;Номер сторінки екрану.
        mov dh,01h    ;Строка
        mov dl,0fh    ;Стовпець.
        int 10h       ;Виклик переривання.
;=====

        ;Вывод второго символа "-"
;-----
        mov ah,0ah
        mov al,45
        mov cx,01h
        int 10h
;=====

        Call Write_day_number

        Call Write_day_of_weak

        Call Write_month_name

        Call Write_year_number

;=====
        ret
;=====
Writte_date Endp

Write_sec Proc
;    Процедура выводит значение текущей секунды.
;-----
        ;Позиціювання курсору.
;-----
        mov ah,02h    ;Код переривання.
        mov bh,00h    ;Номер сторінки екрану.
        mov dh,03h    ;Строка
        mov dl,11h    ;Стовпець.
        int 10h       ;Виклик переривання.
;=====

        ;Получение значения текущего времени:
;-----
        mov ah,2ch
        int 21h
;=====

```

```

;      cl - Значение секунды, которое нужно вывести.
sec_60:      ;Вывод секунды 60
      cmp dh,60
      jl sec_50
;=====
      sub dh,60      ; Вычесть из текущей секунды 60.
      mov dl,dh
      add dl,48      ; Дополнить полученное до ASCII - кода.
      mov ah,02h     ; Выбор функции прерывания.
      int 21h        ; Вызов прерывания.
;=====
      ;Позиціювання курсору.
;-----
      mov ah,02h     ;Код переривання.
      mov bh,00h     ;Номер сторінки екрану.
      mov dh,03h     ;Строка
      mov dl,10h     ;Стовпець.
      int 10h        ;Виклик переривання.
;=====
      mov dl,54      ; Отправить в dl ASCII - код цифры 6.
      mov ah,02h     ; Выбор функции прерывания.
      int 21h        ; Вызов прерывания.
      jmp end_sec

sec_50:      ;Вывод секунды 50-59
      cmp dh,50
      jl sec_40
;=====
      sub dh,50      ; Вычесть из текущей секунды 50.
      mov dl,dh
      add dl,48      ; Дополнить полученное до ASCII - кода.
      mov ah,02h     ; Выбор функции прерывания.
      int 21h        ; Вызов прерывания.
;=====
      ;Позиціювання курсору.
;-----
      mov ah,02h     ;Код переривання.
      mov bh,00h     ;Номер сторінки екрану.
      mov dh,03h     ;Строка
      mov dl,10h     ;Стовпець.
      int 10h        ;Виклик переривання.
;=====
      mov dl,53      ; Отправить в dl ASCII - код цифры 5.
      mov ah,02h     ; Выбор функции прерывания.
      int 21h        ; Вызов прерывания.
      jmp end_sec

sec_40:      ;Вывод секунды 40-49
      cmp dh,40
      jl sec_30
;=====
      sub dh,40      ; Вычесть из текущей секунды 40.

```

```

mov dl,dh
add dl,48      ; Дополнить полученное до ASCII - кода.
mov ah,02h    ; Выбор функции прерывания.
int 21h       ; Вызов прерывания.
;=====
;Позиціювання курсору.
;-----
mov ah,02h    ;Код переривання.
mov bh,00h    ;Номер сторінки екрану.
mov dh,03h    ;Строка
mov dl,10h    ;Стовпець.
int 10h       ;Виклик переривання.
;=====
mov dl,52     ; Отправить в dl ASCII - код цифры 4.
mov ah,02h    ; Выбор функции прерывания.
int 21h       ; Вызов прерывания.
jmp end_sec

sec_30:       ;Вывод секунды 30-39
cmp dh,30
jl sec_20
;=====
sub dh,30     ; Вычесть из текущей секунды 30.
mov dl,dh
add dl,48     ; Дополнить полученное до ASCII - кода.
mov ah,02h    ; Выбор функции прерывания.
int 21h       ; Вызов прерывания.
;=====
;Позиціювання курсору.
;-----
mov ah,02h    ;Код переривання.
mov bh,00h    ;Номер сторінки екрану.
mov dh,03h    ;Строка
mov dl,10h    ;Стовпець.
int 10h       ;Виклик переривання.
;=====
mov dl,51     ; Отправить в dl ASCII - код цифры 3.
mov ah,02h    ; Выбор функции прерывания.
int 21h       ; Вызов прерывания.
jmp end_sec

sec_20:       ;Вывод секунды 20-29
cmp dh,20
jl sec_10
;=====
sub dh,20     ; Вычесть из текущей секунды 20.
mov dl,dh
add dl,48     ; Дополнить полученное до ASCII - кода.
mov ah,02h    ; Выбор функции прерывания.
int 21h       ; Вызов прерывания.
;=====
;Позиціювання курсору.

```

```

;-----
    mov ah,02h    ;Код переривання.
    mov bh,00h    ;Номер сторінки екрану.
    mov dh,03h    ;Строка
    mov dl,10h    ;Стовпець.
    int 10h       ;Виклик переривання.
;=====

    mov dl,50     ; Отправить в dl ASCII - код цифры 2.
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
    jmp end_sec

sec_10:          ;Вывод секунды 10-19
    cmp dh,10
    jl sec_00
;=====
    sub dh,10     ; Вычесть из текущей секунды 10
    mov dl,dh
    add dl,48     ; Дополнить полученное до ASCII - кода.
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
;=====
    ;Позиціювання курсору.
;-----
    mov ah,02h    ;Код переривання.
    mov bh,00h    ;Номер сторінки екрану.
    mov dh,03h    ;Строка
    mov dl,10h    ;Стовпець.
    int 10h       ;Виклик переривання.
;=====
    mov dl,49     ; Отправить в dl ASCII - код цифры 1.
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
    jmp end_sec

sec_00:          ;Вывод секунды 00-09
;=====
    add dh,48     ; Дополнить полученное до ASCII - кода.
    mov dl,dh
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
;=====
end_sec:
;=====
    ret
Write_sec EndP

Write_minute Proc
;    Процедура выводит значение текущей минуты.
;-----
    ;Позиціювання курсору.
;-----

```



```

    mov ah,02h    ;Код переривання.
    mov bh,00h    ;Номер сторінки екрану.
    mov dh,03h    ;Строка
    mov dl,0eh    ;Стовпець.
    int 10h       ;Виклик переривання.
;=====
    ;Получение значения текущего времени:
;-----
    mov ah,2ch
    int 21h
;=====
;    cl - Значение минуты, которое нужно вывести.

Minute_60:      ;Вывод минуты 60
    cmp cl,60
    jl minute_50
;=====
    sub cl,60     ; Вычесть из текущей минуты 60.
    mov dl,cl
    add dl,48     ; Дополнить полученное до ASCII - кода.
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
;=====
    ;Позиціювання курсору.
;-----
    mov ah,02h    ;Код переривання.
    mov bh,00h    ;Номер сторінки екрану.
    mov dh,03h    ;Строка
    mov dl,0dh    ;Стовпець.
    int 10h       ;Виклик переривання.
;=====
    mov dl,54     ; Отправить в dl ASCII - код цифры 6.
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
    jmp end_Minute

Minute_50:      ;Вывод минуты 50-59
    cmp cl,50
    jl minute_40
;=====
    sub cl,50     ; Вычесть из текущей минуты 50.
    mov dl,cl
    add dl,48     ; Дополнить полученное до ASCII - кода.
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
;=====
    ;Позиціювання курсору.
;-----
    mov ah,02h    ;Код переривання.
    mov bh,00h    ;Номер сторінки екрану.
    mov dh,03h    ;Строка
    mov dl,0dh    ;Стовпець.

```

```

        int 10h      ;Виклик переривання.
;=====
        mov dl,53    ; Отправить в dl ASCII - код цифры 5.
        mov ah,02h   ; Выбор функции прерывания.
        int 21h      ; Вызов прерывания.
        jmp end_Minute

Minute_40: ;Вывод минуты 40-49
        cmp cl,40
        jl minute_30
;=====
        sub cl,40     ; Вычесть из текущей минуты 40.
        mov dl,cl
        add dl,48     ; Дополнить полученное до ASCII - кода.
        mov ah,02h   ; Выбор функции прерывания.
        int 21h      ; Вызов прерывания.
;=====
        ;Позиціювання курсору.
;-----
        mov ah,02h   ;Код переривання.
        mov bh,00h   ;Номер сторінки екрану.
        mov dh,03h   ;Строка
        mov dl,0dh   ;Стовпець.
        int 10h      ;Виклик переривання.
;=====
        mov dl,52    ; Отправить в dl ASCII - код цифры 4.
        mov ah,02h   ; Выбор функции прерывания.
        int 21h      ; Вызов прерывания.
        jmp end_Minute

Minute_30: ;Вывод минуты 30-39
        cmp cl,30
        jl minute_20
;=====
        sub cl,30     ; Вычесть из текущей минуты 30.
        mov dl,cl
        add dl,48     ; Дополнить полученное до ASCII - кода.
        mov ah,02h   ; Выбор функции прерывания.
        int 21h      ; Вызов прерывания.
;=====
        ;Позиціювання курсору.
;-----
        mov ah,02h   ;Код переривання.
        mov bh,00h   ;Номер сторінки екрану.
        mov dh,03h   ;Строка
        mov dl,0dh   ;Стовпець.
        int 10h      ;Виклик переривання.
;=====
        mov dl,51    ; Отправить в dl ASCII - код цифры 3.
        mov ah,02h   ; Выбор функции прерывания.
        int 21h      ; Вызов прерывания.
        jmp end_Minute

```

Minute_20: ;Вывод минуты 20-29

cmp cl,20
jl minute_10

=====

sub cl,20 ; Вычесть из текущей минуты 20.
mov dl,cl
add dl,48 ; Дополнить полученное до ASCII - кода.
mov ah,02h ; Выбор функции прерывания.
int 21h ; Вызов прерывания.

=====

;Позиціювання курсору.

mov ah,02h ;Код переривання.
mov bh,00h ;Номер сторінки екрану.
mov dh,03h ;Строка
mov dl,0dh ;Стовпець.
int 10h ;Виклик переривання.

=====

mov dl,50 ; Отправить в dl ASCII - код цифры 2.
mov ah,02h ; Выбор функции прерывания.
int 21h ; Вызов прерывания.
jmp end_Minute

minute_10: ;Вывод минуты 10-19

cmp cl,10
jl minute_00

=====

sub cl,10 ; Вычесть из текущей минуты 10
mov dl,cl
add dl,48 ; Дополнить полученное до ASCII - кода.
mov ah,02h ; Выбор функции прерывания.
int 21h ; Вызов прерывания.

=====

;Позиціювання курсору.

mov ah,02h ;Код переривання.
mov bh,00h ;Номер сторінки екрану.
mov dh,03h ;Строка
mov dl,0dh ;Стовпець.
int 10h ;Виклик переривання.

=====

mov dl,49 ; Отправить в dl ASCII - код цифры 1.
mov ah,02h ; Выбор функции прерывания.
int 21h ; Вызов прерывания.
jmp end_Minute

Minute_00: ;Вывод минуты 00-09

=====

add cl,48 ; Дополнить полученное до ASCII - кода.
mov dl,cl
mov ah,02h ; Выбор функции прерывания.

```

        int 21h      ; Вызов прерывания.
;=====

End_Minute:
;=====
        ret
Write_minute EndP

Write_Hour Proc
;      Процедура выводит значение текущего часа.
;-----
;Позиціювання курсору.
;-----
        mov ah,02h   ;Код переривання.
        mov bh,00h   ;Номер сторінки екрану.
        mov dh,03h   ;Строка
        mov dl,0bh   ;Стовпець.
        int 10h      ;Виклик переривання.
;=====

;Получение значения текущего времени:
;-----
        mov ah,2ch
        int 21h
;=====
;      ch - Значение часа, которое нужно вывести.
hour_20:      ;Вывод часа 20-24
        cmp ch,20
        jl hour_10
;=====
        sub ch,20     ; Вычесть из текущего часа 20
        mov dl,ch
        add dl,48     ; Дополнить полученное до ASCII - кода.
        mov ah,02h   ; Выбор функции прерывания.
        int 21h      ; Вызов прерывания.
;=====
;Позиціювання курсору.
;-----
        mov ah,02h   ;Код переривання.
        mov bh,00h   ;Номер сторінки екрану.
        mov dh,03h   ;Строка
        mov dl,0ah   ;Стовпець.
        int 10h      ;Виклик переривання.
;=====
        mov dl,50     ; Отправить в dl ASCII - код цифры 2.
        mov ah,02h   ; Выбор функции прерывания.
        int 21h      ; Вызов прерывания.
        jmp end_hour

hour_10:      ;Вывод часа 10-19
        cmp ch,10
        jl hour_00

```

```

;=====
    sub ch,10      ; Вычесть из текущего часа 10
    mov dl,ch
    add dl,48      ; Дополнить полученное до ASCII - кода.
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
;=====
    ;Позиціювання курсору.
;-----
    mov ah,02h    ;Код переривання.
    mov bh,00h    ;Номер сторінки екрану.
    mov dh,03h    ;Строка
    mov dl,0ah    ;Стовпець.
    int 10h       ;Виклик переривання.
;=====
    mov dl,49     ; Отправить в dl ASCII - код цифры 1.
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
    jmp end_hour

hour_00:         ;Вывод часа 00-09
;=====
    add ch,48     ; Дополнить полученное до ASCII - кода.
    mov dl,ch
    mov ah,02h    ; Выбор функции прерывания.
    int 21h       ; Вызов прерывания.
;=====

end_hour:

;=====
    ret
Write_Hour EndP

Writte_time Proc
;-----
    ;Позиціювання курсору.
;-----
    mov ah,02h    ;Код переривання.
    mov bh,00h    ;Номер сторінки екрану.
    mov dh,03h    ;Строка
    mov dl,01h    ;Стовпець.
    int 10h       ;Виклик переривання.
;=====
    ;Вивід строки
;-----
    mov dx,offset time
    mov ah,09h    ;Код переривання.
    int 21h       ;Виклик переривання.
;=====

    ;Вывод нулевого значения времени.

```

```

;-----
;Позиціювання курсору на позицію.
;-----
mov ah,02h ;Код переривання.
mov bh,00h ;Номер сторінки екрану.
mov dh,03h ;Строка
mov dl,0ah ;Стовпець.
int 10h ;Виклик переривання.
;=====
;Вывод восьми цифр 00
;-----
mov ah,0ah
mov al,48
mov cx,08h
int 10h
;=====
;Позиціювання курсору на позицію.
;-----
mov ah,02h ;Код переривання.
mov bh,00h ;Номер сторінки екрану.
mov dh,03h ;Строка
mov dl,0ch ;Стовпець.
int 10h ;Виклик переривання.
;=====
;Вывод символа ":"
;-----
mov ah,0ah
mov al,58
mov cx,01h
int 10h
;=====
;Позиціювання курсору на позицію.
;-----
mov ah,02h ;Код переривання.
mov bh,00h ;Номер сторінки екрану.
mov dh,03h ;Строка
mov dl,0fh ;Стовпець.
int 10h ;Виклик переривання.
;=====
;Вывод символа "."
;-----
mov ah,0ah
mov al,46
mov cx,01h
int 10h
;=====

call Write_Hour
call Write_Minute
Call Write_sec

ret

```

```

;=====
Write_time Endp

Read_hour Proc
;   Процедура получения нового значения часа.
;-----
;   ;Позиціювання курсору.
;-----
;       mov ah,02h   ;Код переривання.
;       mov bh,00h   ;Номер сторінки екрану.
;       mov dl,01h   ;Стовпець.
;       mov dh,0fh   ;Строка
;       int 10h      ;Виклик переривання.
;=====
;   Чтение символа с клавиатуры.
;-----
;   al - Полученный символ.
;       mov ah,01h   ; Код функции прерывания.
;       int 21h      ; Вызов функции.
;=====

;       mov bl,10    ; Установка второго операнда.
;       mul bl       ; Умножение полученного числа на 10. (Первый операнд хранится в al.)
;       mov New_hour,al ; Сохранение первой полученной цифры.

;   Чтение символа с клавиатуры.
;-----
;       mov ah,01h   ; Код функции прерывания.
;       int 21h      ; Вызов функции.
;=====

;       add al,New_hour ; Сохранение первой полученной цифры.
;       mov New_hour,al ; Полученного числа.
;=====
;=====
ret
Read_hour EndP

```

```

Read_day Proc
;   Процедура получения нового значения дня.
;-----
;   ;Позиціювання курсору.
;-----
;       mov ah,02h   ;Код переривання.
;       mov bh,00h   ;Номер сторінки екрану.
;       mov dh,01h   ;Строка
;       mov dl,0ah   ;Стовпець.
;       int 10h      ;Виклик переривання.
;=====
;   Чтение символа с клавиатуры.
;-----
;   al - Полученный символ.

```

```

        mov ah,01h    ; Код функции прерывания.
        int 21h       ; Вызов функции.
;=====

        mov dl,10     ; Установка второго операнда.
        sub al,30h     ; Уменьшить полученное ASCII - значение до цифры.
        mul dl         ; Умножение полученного числа на 10. (Первый операнд хранится в al.)
        mov New_day,al ; Сохранение первой полученной цифры.

;      Чтение символа с клавиатуры.
;-----
        mov ah,01h    ; Код функции прерывания.
        int 21h       ; Вызов функции.
;=====

        sub al,30h     ; Уменьшить полученное ASCII - значение до цифры.
        add al,New_day ; Сохранение первой полученной цифры.
        mov New_day,al ; Полученного числа.
;=====
;=====
        ret
Read_day EndP

Read_month Proc
;      Процедура получения нового значения месяца.
;      Позиціювання курсору.
;-----
        mov ah,02h    ;Код переривання.
        mov bh,00h    ;Номер сторінки екрану.
        mov dl,0dh    ;Стовпець.
        mov dh,01h    ;Строка
        int 10h       ;Виклик переривання.
;=====
;      Чтение символа с клавиатуры.
;-----
;      al - Полученный символ.
        mov ah,01h    ; Код функции прерывания.
        int 21h       ; Вызов функции.
;=====

        mov dl,10     ; Установка второго операнда.
        sub al,30h     ; Уменьшить полученное ASCII - значение до цифры.
        mul dl         ; Умножение полученного числа на 10. (Первый операнд хранится в al.)
        mov New_month,al ; Сохранение первой полученной цифры.

;      Чтение символа с клавиатуры.
;-----
        mov ah,01h    ; Код функции прерывания.
        int 21h       ; Вызов функции.
;=====

        sub al,30h     ; Уменьшить полученное ASCII - значение до цифры.
        add al,New_month ; Сохранение первой полученной цифры.
        mov New_month,al ; Полученного числа.

```



```

;=====
;=====
ret
Read_month EndP

Read_year Proc
;    Процедура получения нового значения года.
;    ;Позиціювання курсору.
;-----
    mov ah,02h    ;Код переривання.
    mov bh,00h    ;Номер сторінки екрану.
    mov dl,20     ;Стовпець.
    mov dh,01h    ;Строка
    int 10h       ;Виклик переривання.
;=====
;    Чтение символа с клавиатуры.
;-----
;    al - Полученный символ.
    mov ah,01h    ; Код функции прерывания.
    int 21h       ; Вызов функции.
;=====
    mov dl,10     ; Установка второго операнда.
    sub al,30h    ; Уменьшить полученное ASCII - значение до цифры.
;    mul dl       ; Умножение полученного числа на 10. (Первый операнд хранится в al.)
    mov New_year_to_write,al ; Полученного числа.
;=====
;=====
ret
Read_year EndP

Zapit_na_izmenenie Proc
Zapit_na_izmenenie_begin:

;    Процедура, выводящая запрос на изменение даты и/или времени.
;-----
;    ;Позиціювання курсору.
;-----
    mov ah,02h    ;Код переривання.
    mov bh,00h    ;Номер сторінки екрану.
    mov dl,01h    ;Стовпець.
    mov dh,09h    ;Строка
    int 10h       ;Виклик переривання.
;=====
;    ;Вивід строки меню.
;-----
;    ;Вивід строки
    mov dx,offset Zapros_i_can_do
    mov ah,09h    ;
    int 21h       ;
;=====
;    ;Позиціювання курсору.
;-----

```

```

        mov ah,02h    ;Код переривання.
        mov bh,00h    ;Номер сторінки екрану.
        mov dl,01h    ;Стовпець.
        mov dh,0ah    ;Строка
        int 10h       ;Виклик переривання.
;=====
;-----
        mov dx,offset Zaprosh_Data
        mov ah,09h    ;
        int 21h       ;
;=====
;-----
        ;Позиціювання курсору.
;-----
        mov ah,02h    ;Код переривання.
        mov bh,00h    ;Номер сторінки екрану.
        mov dl,01h    ;Стовпець.
        mov dh,0bh    ;Строка
        int 10h       ;Виклик переривання.
;=====
;-----
        mov dx,offset Zaprosh_Time
        mov ah,09h    ;
        int 21h       ;
;=====
;-----
        ;Позиціювання курсору.
;-----
        mov ah,02h    ;Код переривання.
        mov bh,00h    ;Номер сторінки екрану.
        mov dl,01h    ;Стовпець.
        mov dh,0ch    ;Строка
        int 10h       ;Виклик переривання.
;=====
;-----
        mov dx,offset Zaprosh_refersh
        mov ah,09h    ;
        int 21h       ;
;=====
;-----
        ;Позиціювання курсору.
;-----
        mov ah,02h    ;Код переривання.
        mov bh,00h    ;Номер сторінки екрану.
        mov dl,01h    ;Стовпець.
        mov dh,0eh    ;Строка
        int 10h       ;Виклик переривання.
;=====
;-----
        mov dx,offset Zaprosh_exit
        mov ah,09h    ;
        int 21h       ;
;=====
;-----
        ;Позиціювання курсору.
;-----

```

```

mov ah,02h ;Код переривання.
mov bh,00h ;Номер сторінки екрану.
mov dl,01h ;Стовпець.
mov dh,0fh ;Строка
int 10h ;Виклик переривання.

```

```

;=====

```

```

; Чтение символа с клавиатуры.

```

```

;-----

```

```

mov ah,01h ; Код функции прерывания.
int 21h ; Вызов функции.

```

```

; al - Полученный символ.

```

```

;=====

```

```

cmp al,48
je Zapit_exit
cmp al,49
je Zapit_change_date
cmp al,50
je Zapit_change_time
cmp al,51
je Zapit_refresh

```

```

jmp Zapit_exit

```

```

Zapit_change_date:

```

```

call Ochistka_Ecrana

```

```

call Podscaska_Date

```

```

;Вывод нулевого значения даты.

```

```

;-----

```

```

;Позиціювання курсору на позицію.

```

```

;-----

```

```

mov ah,02h ;Код переривання.
mov bh,00h ;Номер сторінки екрану.
mov dh,01h ;Строка
mov dl,0ah ;Стовпець.
int 10h ;Виклик переривання.

```

```

;=====

```

```

;Вывод восьми цифр 00

```

```

;-----

```

```

mov ah,0ah
mov al,48
mov cx,08h
int 10h

```

```

;=====

```

```

;Позиціювання курсору на позицію.

```

```

;-----

```

```

mov ah,02h ;Код переривання.
mov bh,00h ;Номер сторінки екрану.
mov dh,01h ;Строка
mov dl,0ch ;Стовпець.
int 10h ;Виклик переривання.

```

```

;=====
;Вывод первого символа "-"
;-----
mov ah,0ah
mov al,45
mov cx,01h
int 10h
;=====
;Позиціювання курсору на позицію.
;-----
mov ah,02h ;Код переривання.
mov bh,00h ;Номер сторінки екрану.
mov dh,01h ;Строка
mov dl,0fh ;Стовпець.
int 10h ;Виклик переривання.
;=====
;Вывод второго символа "-"
;-----
mov ah,0ah
mov al,45
mov cx,01h
int 10h
;=====

call Read_day
call Read_month

; Установка нового значения времени.
;-----
mov dl,new_day ;\
mov dh,new_month;-- Установка новых значений даты по регистрам
mov cx,New_year ;/ для установки в системе через прерывание.
mov ah,2bh ; Передача кода прерывания.
int 21h ; Вызов прерывания.
;=====

call read_year

;Вывод вывод последнего символа года - 3.
;-----
mov ah,0ah
mov al,new_year_to_write
mov cx,01h
int 10h
;=====
cmp al,0
; jne Prn_error_date
jmp zapit_exit

Prn_error_date:
;Вивід строки
;-----

```

```

    mov dx,offset error
    mov ah,09h      ;Код переривання.
    int 21h        ;Виклик переривання.
;=====
    jmp Zapit_exit

Zapit_change_time:
    call Ochistka_Ecrana
    call Podscaska_time
    call Read_hour
    jmp Zapit_exit

Zapit_refresh:
    call Ochistka_Ecrana
    call Podscaska_Date
    call Podscaska_time
    call Writte_date
    call Writte_time
    call My_Copyright
    jmp Zapit_na_izmenenie_begin

Zapit_exit:

    ret
;=====
Zapit_na_izmenenie EndP

My_Copyright Proc
;-----
    ;Позиціювання курсору.
;-----
    mov ah,02h      ;Код переривання.
    mov bh,00h      ;Номер сторінки екрану.
    mov dl,01h      ;Стовпець.
    mov dh,15h      ;Строка
    int 10h         ;Виклик переривання.
;=====
;-----
    ;Вивід строки копирайт.
    mov dx,offset string
    mov ah,09h      ;
    int 21h         ;
;=====
    ;Позиціювання курсору.
;-----
    mov ah,02h      ;Код переривання.
    mov bh,00h      ;Номер сторінки екрану.
    mov dl,01h      ;Стовпець.
    mov dh,10h      ;Строка
    int 10h         ;Виклик переривання.
;=====
    ret

```

```
;=====
My_Copyright endp
```

begin:

```
mov ax,@data
mov ds,ax
```

```
call Ochistka_Ecrana
call Podscaska_Date
call Podscaska_time
call Writte_date
call Writte_time
call zapit_na_izmenenie
```

```
call My_Copyright
```

```
mov ax,4c00h
int 21h
end begin
```

```
*****
```

```
INT 21 f.2D - Установить системное время
```

```
AH = 2Dh
```

```
CH = часы
```

```
CL = минуты
```

```
DH = секунды
```

```
DL = 1/100 секунд
```

```
Возвращает: AL = результат
```

```
00h операция завершена успешно
```

```
FFh ошибка
```

```
*****
```