Лимаренко Александр, группа БПИ192

Вариант 13

Разработать программу вычисления даты каталической Пасхалии для заданного года.

Код программы:

```
format PE console
include 'win32a.inc'
entry start

section 'data' data readable writeable

        str1 db 'Enter year:',10,0          ;строки для вывода
        str2 db 'Computus date is %d ',0
        str3 db 'March',10,0         ;в случае разных месяцев
        str4 db 'April',10,0
        str5 db 'Number must be positive',10,0
        scanf_int db '%d',0          ;для ввода чисел в scanf
        year dd ?                   ;переменная для года
        month dd ?                    ;месяца
        date dd ?                   ;дня

         Y dd ?      ;переменные для алгоритма вычисления дня пасхи
         G dd ?
         C dd ?
         X dd ?
         Z dd ?
         D dd ?
         E dd ?
         N dd ?

section 'text' code executable readable
start:                  ;начало программы
        push str1
        call [printf]    ;вывод начальной строки
        push year        ;в перменную year
        push scanf_int   ;тип - int (dword)
        call [scanf]     ;вызов функции scanf

        xor eax,eax      ;eax=0
        cmp [year],eax   ;сравниваем year и eax(=0)
        jle .incorrect   ;если меньше или равно то перейти к метке .incorrect (знаковое сравнение)


        call GetComputusDate   ;вызов функции для вычисления даты пасхи
        push [date]        ;переменая даты
        push str2          ;в строке str2
        call [printf]          ;вывод строки str2 с date место "%d" с помошью printf
        mov ebx,str4       ;если переменая month равна 0 то ebx = str3(март) если month = 1 то ebx =
str4(Апрель)
        mov eax,[month]
        cmp eax,1
        je .april
```

```
 mov ebx,str3
    .april:
    push ebx        ;вывод строки ebx(str3 или str4)
    call [printf]

    jmp .end        ;пропустить вывод сообщение о некорректности данных

    .incorrect:
    push str5       ;вывод сообщение о некорректности данных
    call [printf]

    .end:
    call [getch]    ;ожидаем нажатия любой клавиши
    ret


GetComputusDate:        ;алгоритм вычисления даты пасхи

    mov eax,[year]    ;записываем текущий год в  переменную Y
    mov [Y],eax

    mov eax,[Y]       ;G = (Y mod 19) + 1
    xor edx,edx       ;подготавливаемся к делению
    mov ebx,19        ;для вычисления остатка
    div ebx           ;берем остаток от 19
    mov eax,edx       ;остаток оставляется в edx
    inc eax           ; +1
    mov [G],eax       ;сохраняем переменную G

    mov eax,[Y]       ;C = Y/100 + 1
    xor edx,edx       ;подготавливаемся к делению
    mov ebx,100       ;делим на 100
    div ebx           ;делим
    inc eax           ;+1
    mov [C],eax       ;сохраняем

    mov eax,[C]       ;X = 3C/4 - 12
    mov ebx,3         ;надо умножить на 3
    mul ebx           ;умножаем
    shr eax,2         ;битовый сдвиг вправо на 2 бита - равноценно делению на 4
    sub eax,12        ;-12
    mov [X],eax       ;сохраняем

    mov eax,[C]       ;Z = (8C + 5)/25 - 5
    shl eax,3         ;ботовый сдвиг влево на 3 бита - равноценно умножению на 8
    add eax,5         ;+1
    xor edx,edx       ;подготавливаемся к делению
    mov ebx,25        ;требуется разделить на 25
    div ebx           ;деление
    sub eax,5         ;-5
    mov [Z],eax       ;сохраняем
```

```
mov eax,[Y]       ;D = 5Y/4 - X - 10
    mov ebx,5         ;требуется умножить на 5
    mul ebx           ;умножение
    shr eax,2         ;битовый сдвиг вправо на 2 бита = деление на 4
    sub eax,[X]       ;-X
    sub eax,10        ;-10
    mov [D],eax       ;сохраняем

    mov eax,[G]       ;[(11G + 20 + Z - X) mod 30 + 30] mod 30
    mov ebx,11        ;подготавливаемся к умножению на 11
    mul ebx           ;умножаем
    add eax,20        ;+20
    add eax,[Z]       ;+Z
    sub eax,[X]       ;+X
    mov ebx,30        ;подготавливаемся к делению на 30
    xor edx,edx
    div ebx           ;деление
    mov eax,edx       ;остаток перемещаем из edx в eax
    add eax,30        ;+30
    xor edx,edx       ;подготавливаемся к делению
    div ebx           ;деление
    mov [E],edx       ;сохраняем ответ

    cmp [E],24        ;ЕСЛИ (E = 24) ИЛИ (E = 25 И G > 11), ТО увеличить E на 1
    je .l1            ;если E=24 то inc [E]
    cmp [E],25        ;если E не равно 25 то пропустить увеличение
    jne .l2
    cmp [G],11        ;если G меньше или равно 11 то пропустить увеличение
    jle .l2
    .l1:
    inc [E]
    .l2:

    mov eax,44        ;N = 44 - E
    sub eax,[E]       ;-E
    mov [N],eax       ;сохраняем
```

```
 cmp [N],21        ;ЕСЛИ N < 21, ТО увеличить N на 30
      jge .l3          ;если N больше или равно 30 то пропустить увеличение
      add [N],30
      .l3:

      mov eax,[N]       ;N = N + 7 - (D + N) mod 7
      mov ecx,eax       ;сохраняем копию N в ecx
      add eax,7         ;+7
      mov edi,eax       ;сохраняем результат суммы в edi
      mov eax,ecx       ;eax=N
      add eax,[D]       ;+D
      xor edx,edx       ;подготавливаемся к делению
      mov ebx,7         ;на 7
      div ebx           ;деление
      sub edi,edx       ;вычитаем из "N+7" "(D+N) mod 7"
      mov [N],edi       ;сохраняем результат

      cmp [N],31        ;ЕСЛИ N > 31, ТО дата Пасхи (N - 31) апреля, ИНАЧЕ дата Пасхи N марта
      jbe .l4           ;если N мельше или равно 31 то пропустить "переключение" на апрель
      sub [N],31
      mov [month],1
      .l4:

      mov eax,[N]       ;сохранение результатов функции
      mov [date],eax
      ret

  section '.idata' import readable

  library msvcrt,'msvcrt.dll'

  import msvcrt,\
      printf,'printf',\
      scanf,'scanf',\
      getch,'_getch'
```

, Алгоритм расчета взят с Википедии: https://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D1%81%D1%85%D0%B0%D0%BB%D0%B8%D1%8F

В программе присутствует базовая проверка на неправильный ввод данных.
Например, не получится найти день пасхи для года <1