|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

**ФАКУЛЬТЕТ** ***ИУК «Информатика и управление»***

**КАФЕДРА** \_\_***ИУК4 «Программное обеспечение ЭВМ, информационные технологии»***

**ЛАБОРАТОРНАЯ РАБОТА №5**

**«Программирование ветвлений»**

**ДИСЦИПЛИНА: «Машинно-зависимые языки программирования»**

|  |  |  |
| --- | --- | --- |
| Выполнил: студент гр. ИУК4-31Б | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Отрошенко Т. В.)  (Подпись) |
| Проверил: | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Амеличева К. А.)  (Подпись) |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | | |
|  | - Балльная оценка:  - Оценка: | |

Калуга, 2021

**Цель работы:** практическое овладение навыками разработки программного кода на языке Ассемблер. Изучение команд условного и безусловного перехода. Исследование организации переходов.

**Задачи:** разработка простой программы, использующей операторы передачи управления и приемов программирования арифметических выражений, содержащих разветвления.

**Задание 1 (Вариант 20)**

**** ****

|  |  |
| --- | --- |
| .model small  .stack 100h  .data  a db -15  x db 5  y dw 0  .code  start:  mov ax, @data  mov ds, ax  mov ax, 0  cmp x, 5 ; первая часть выражения  jnl y1\_2  mov al, x  sal al, 1 ; умножение на два  jmp sec\_  y1\_2:  mov al, a  cmp al, 0  js abs\_ | abs\_rtn:  add al, x ; сложение a и x  jmp sec\_  abs\_: ; модуль числа  xor al, 0FFh ;? 01111111b  add al, 1  jmp abs\_rtn  sec\_: ; вторая часть выражения  xchg ax, bx ; сохранение результата  cmp x, 0  jz y2\_2  cmp a, 0 ; поместить байт в слово с сохранением знака  jns z\_sign  mov ah, 0FFh  z\_sign:  mov al, a  idiv x  jmp thr\_ |

|  |  |
| --- | --- |
| y2\_2:  mov al, 4  jmp thr\_  thr\_: ; третья часть выражения  imul bl  mov y, ax | mov ax,4c00h  int 21h  end start  Результат: |

**Задание 2**

2. Составить программу арифметических и логических действий над целыми переменными и константами.

3. Сформировать меню, предлагающее пользователю сделать выбор значения диапазона. Для ввода символа используется функция DOS 01h (при вводе символ отображается на экране).

1. В центре чистого экрана: сформировать меню отражающее суть задания и предлагающее пользователь ввести символ 1,2 или 3 для расчета соответствующего выражения при заданных значениях переменных a и b.

2. Вывести по углам экрана:

* Фамилию автора программы;
* Номер группы;
* Факультет;
* Символ ‘!’, повторив его 5 раз.

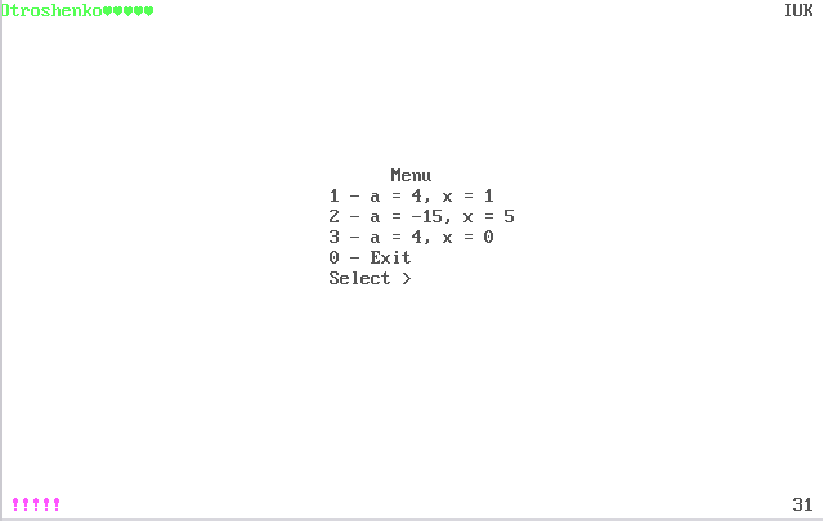
3. Задать видеоатрибуты, определяющие цвет символов и цвет фона при выводе символов на экран. Выбор цветового оформления зависит от вкуса.

|  |  |
| --- | --- |
| .model small  .stack 100h  .data  nam db 'Otroshenko$'  fk db 'IUK$'  gr db '31$'  s db '!!!!!$' | menu db 'Menu',13,10  db 09, 09, 09, 09,'1 - a = 4, x = 1',13,10  db 09, 09, 09, 09,'2 - a = -15, x = 5', 13,10  db 09, 09, 09, 09,'3 - a = 4, x = 0', 13,10  db 09, 09, 09, 09,'0 - Exit', 13,10,'$'  select db 09, 09, 09, 09,'Select > $' |

|  |  |  |
| --- | --- | --- |
| output db 'Answer ready $'  a db 4, -15, 4  x db 1, 5, 0  y dw 0  .code  start:  mov ax,@data  mov ds, ax  ; установка видеорежима  mov ah, 0fh  int 10h  ; mov ah, 0Bh  ; mov dl, 0E3h  ; int 10h  mov ah, 00  int 10h  ;вывод информации по углам  ; лев верх  mov ah, 02  mov dh, 0  mov dl, 1  int 10h  mov ah, 09  lea dx, nam  int 21h | ; вниз  mov ah, 02  mov dh, 24  mov dl, 1  int 10h  mov ah, 09  lea dx, s  int 21h  ; право верх  mov ah, 02  mov dh, 0  mov dl, 76  int 10h  mov ah, 09  lea dx, fk  int 21h  ; вниз  mov ah, 02  mov dh, 24  mov dl, 77  int 10h  mov ah, 09  lea dx, gr  int 21h  ; меню  mov ah, 02  mov dh, 8  mov dl, 38  int 10h | mov ah, 09h  lea dx, menu  int 21h  select\_loop:  mov ah,09h  lea dx, select  int 21h  mov ah,01h  int 21h  mov bl, al  sub bl, 31h  mov di, offset a  mov si, offset x  mov ch, si[bx]; - для х  mov cl, di[bx]; - для a  cmp al,'0'  jz exit  cmp al,'1'  jz cal  cmp al,'2'  jz cal  cmp al,'3'  jz cal  jmp select\_loop |

|  |  |
| --- | --- |
| cal: ; взято из prog5\_2 с заменой а на di[bx], x на si[bx] и bl на dl  mov ax, 0  cmp ch, 5 ; первая часть выражения  jnl y1\_2  mov al, ch  sal al, 1 ; умножение на два  jmp sec\_  y1\_2:  mov al, cl  cmp al, 0  js abs\_  abs\_rtn:  add al, ch ; сложение a и x  jmp sec\_  abs\_: ; модуль числа  xor al, 0FFh  add al, 1  jmp abs\_rtn  sec\_: ; вторая часть выражения  xchg ax, dx ; сохранение результата  cmp ch, 0  jz y2\_2 | cmp cl, 0 ; поместить байт в слово с сохранением знака  jns z\_sign  mov ah, 0FFh  z\_sign:  mov al, cl  idiv ch  jmp thr\_    y2\_2:  mov al, 4  jmp thr\_  thr\_: ; третья часть выражения  imul dl  mov y, ax  ;вывод ответа  mov ah, 02  mov dh, 14  mov dl, 32  int 10h  mov ah, 09h  lea dx, output  int 21h  exit:  mov ax,4c00h  int 21h  end start |

**Окно консоли:**





**Результат выполнения программы:**

При закуске пункта 1 =  = 8

При закуске пункта 2 =  = -60

При закуске пункта 3 =  = 0

**Вывод:** в ходе выполнения лабораторной работы было проведено практическое овладение навыками разработки программного кода на языке Ассемблер. Изучение команд условного и безусловного перехода. Исследование организации переходов.