|  |  |
| --- | --- |
| 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 №7**

**«Работа с файлами под управлением MS DOS.**

**Функции файловой системы»**

**ДИСЦИПЛИНА: «Системное программирование»**



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Выполнил: студент гр. ИУК4-31Б | |  |  | ( | Суриков Н. С. | ) |
|  |  |  | (подпись) |  | (Ф.И.О.) |  |
| Проверил: | |  |  | ( | Амеличева К. А. | ) |
|  |  |  | (подпись) |  | (Ф.И.О.) |  |

|  |  |
| --- | --- |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | |
|  | - Балльная оценка:  - Оценка: |

**Цель:** реализовать основные операции работы с файлами: открытие файла для чтения/записи, ввод-вывод в файл и т.д. Научиться использовать цепочечные команды.

**Вариант №20**

В каждой четной строке произвольного текстового файла изменить последовательность символов на обратную. Исходные и измененные строки показать на экране. Изменения сохранить в новом файле.

**Теоретическая часть:**

stosb — Сохранение байта из регистра al в строке

lodsb — Сохранение байта из строки в регистре al

cld — Изменение направления прохода по строке (вверх)

std - Изменение направления прохода по строке (вниз)

**Листинг программы:**

1 .model small

2 .stack 100h

3 .386

4 .data

5 input\_file\_name db "infile.txt", 0

6 output\_file\_name db "outfile.txt", 0

7

8 input\_file\_ID dw ?

9 output\_file\_ID dw ?

10

11 opening\_error\_message db "File could not be opened", "$"

12 creation\_error\_message db "File could not be created", "$"

13 read\_error\_message db "File could not be read", "$"

14 write\_error\_message db "Error in writing to file", "$"

15 completion\_message db "Program completed successfully", "$"

16

17 newline db 0Dh, 0Ah, 0 *; Символы конца строки (CRLF)*

18

19 char\_buffer db 1 dup(0) *; Буфер для чтения символа*

20 line\_buffer db 255 dup(0) *; Буфер для текущей строки*

21 line\_length dw 0 *; Длина текущей строки*

22 line\_number dw 1 *; Счётчик строк*

23

24 reverse\_buffer db 255 dup(0) *; Буфер для реверсированной строки*

25

26 .code

27 mPrint macro string

28 mov ah, 09h

29 lea dx, string

30 int 21h

31 endm

32

33 start:

34 *; Инициализация сегмента данных*

35 mov ax, @data

36 mov ds, ax

37 mov es, ax

38

39 *; Открытие входного файла*

40 mov ah, 3Dh

41 xor al, al

42 lea dx, input\_file\_name

43 int 21h

44 jc opening\_error

45 mov input\_file\_ID, ax

46

47 *; Создание выходного файла*

48 mov ah, 3Ch

49 xor cx, cx

50 lea dx, output\_file\_name

51 int 21h

52 jc creation\_error

53 mov output\_file\_ID, ax

54

55 file\_read\_loop:

56 *; Чтение символа из файла*

57 mov ah, 3Fh

58 mov bx, input\_file\_ID

59 lea dx, char\_buffer

60 mov cx, 1 *; Читаем 1 байт*

61 int 21h

62 jc read\_error

63 cmp ax, 0 *; Конец файла?*

64 je end\_of\_file

65

66 *; Проверка на конец строки*

67 mov al, char\_buffer *; Получаем прочитанный символ*

68 cmp al, 0Dh *; Возврат каретки?*

69 je end\_of\_line

70 cmp al, 0Ah *; Перевод строки?*

71 je end\_of\_line

72

73 *; Сохранение символа в буфер строки*

74 lea di, line\_buffer

75 mov cx, line\_length

76 add di, cx *; Указатель на конец строки*

77 mov [di], al *; Сохраняем символ*

78 inc word ptr line\_length *; Увеличиваем длину строки*

79 jmp file\_read\_loop

80

81 end\_of\_line:

82 *; Добавляем символы конца строки в конец строки*

83 lea di, line\_buffer

84 add di, line\_length

85 mov al, 0Dh

86 stosb

87 mov al, 0Ah

88 stosb

89 mov al, '$'

90 stosb

91

92 *; Выводим строку на экран*

93 lea dx, line\_buffer

94 mPrint line\_buffer

95

96 *; Строка завершена, проверяем её чётность*

97 mov ax, line\_number

98 test ax, 1

99 jnz skip\_reverse *; Нечётная строка — пропускаем*

100

101 *; Реверсируем чётную строку*

102 lea si, line\_buffer

103 lea di, reverse\_buffer

104 add si, line\_length

105 dec si

106 mov cx, line\_length

107 reverse\_loop:

108 std *; Устанавливаем направление*

109 lodsb *; Читаем символ из строки*

110 cld *; Меняем направление*

111 stosb *; Записываем в реверсированный буфер*

112 loop reverse\_loop

113

114 *; Добавляем символы конца строки (CRLF) в конец строки*

115 lea di, reverse\_buffer

116 add di, line\_length

117 mov al, 0Dh

118 stosb

119 mov al, 0Ah

120 stosb

121 mov al, '$'

122 stosb

123

124 *; Выводим перевёрнутую строку на экран*

125 lea dx, reverse\_buffer

126 mPrint reverse\_buffer

127

128 *; Записываем реверсированную строку в выходной файл*

129 lea dx, reverse\_buffer

130 mov cx, line\_length

131 call write\_line

132 jmp continue\_processing

133

134 skip\_reverse:

135 *; Записываем исходную строку в выходной файл*

136 lea dx, line\_buffer

137 mov cx, line\_length

138 call write\_line

139

140 continue\_processing:

141 *; Увеличение номера строки и сброс длины*

142 inc word ptr line\_number

143 mov word ptr line\_length, 0

144 *; Добавляем конец строки в файл вручную*

145 lea dx, newline

146 mov ah, 40h

147 mov bx, output\_file\_ID

148 lea dx, newline

149 mov cx, 2

150 int 21h

151 jmp file\_read\_loop

152

153 end\_of\_file:

154 *; Закрытие файлов*

155 mov ah, 3Eh

156 mov bx, input\_file\_ID

157 int 21h

158

159 mov ah, 3Eh

160 mov bx, output\_file\_ID

161 int 21h

162

163 *; Успешное завершение*

164 mPrint completion\_message

165 jmp end\_program

166

167 write\_line:

168 *; Записываем строку в файл*

169 mov ah, 40h

170 mov bx, output\_file\_ID

171 int 21h

172 jc write\_error

173 ret

174

175 opening\_error:

176 mPrint opening\_error\_message

177 jmp end\_program

178

179 creation\_error:

180 mPrint creation\_error\_message

181 jmp end\_program

182

183 read\_error:

184 mPrint read\_error\_message

185 jmp end\_program

186 write\_error:

187 mPrint write\_error\_message

188 jmp end\_program

189

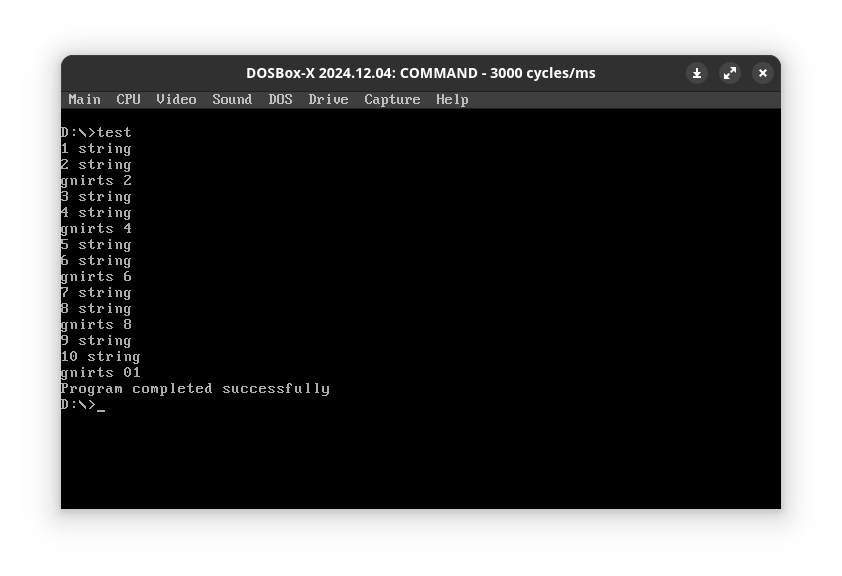
190 end\_program:

191 mov ax, 4C00h

192 int 21h

193 end start

**Результат выполнения программы:**

****

|  |  |
| --- | --- |
| **infile.txt:** | **outfile.txt:** |
| 1 string  2 string  3 string  4 string  5 string  6 string  7 string  8 string  9 string  10 string | 1 string  gnirts 2  3 string  gnirts 4  5 string  gnirts 6  7 string  gnirts 8  9 string  gnirts 01 |

**Вывод:** в ходе выполнения работы были сформированы практические навыки разработки программного кода на языке Ассемблера; изучены способы работы с файлами, цепочечными командами.