Федеральное государственное автономное

образовательное учреждение

Высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

|  |
| --- |
| Институт космических и информационных технологий |
| институт |
| Программная инженерия |
| кафедра |

**ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ**

|  |
| --- |
| Структуры и работа с файлами |
| тема |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Преподаватель | |  |  |  | М. В. Сарамуд |
|  | |  | подпись, дата |  | инициалы, фамилия |
| Студент | КИ21-17/1Б, 032156940 |  |  |  | Н. А. Самарин |
|  | номер группы, зачётной книжки |  | подпись, дата |  | инициалы, фамилия |

Красноярск 2024

**СОДЕРЖАНИЕ**

1 Задание............................................................................................................... 3

2 Вариант.............................................................................................................. 3

3 Ход работы........................................................................................................ 3

4 Вывод................................................................................................................. 6

**1 Задание**

Разработать программу на Ассемблере, которая содержит структуру,  
описанную в варианте, создает экземпляры структуры (не менее 3) и считывает  
значения всех полей из входящего файла (файл можно создавать как  
программой на ассемблере, так и любыми средствами, файл содержит значения  
полей вашей структуры, соответственно, не менее 3 экземпляров), код  
основной программы не содержит значений полей структуры, которые  
читаются из файла. Далее необходимо изменить одно из значений у всех  
экземпляров структуры и записать все поля структуры в новый файл. Если  
выходной файл получается сложно читаемым для человека, можно дублировать  
в консоль всю запись.

**2 Вариант**

Вариант 24. Структура билет, содержит поля Пункт назначения, Пункт  
вылета, Дата, Стоимость, прибавить к стоимости +99.

**3 Ход работы**

В работе был использован FASM на операционной системе Windows 10.  
Ниже представлен код разработанных программ и скриншоты их работы.

Листинг 1 – Код программы

format PE Console  
entry start  
include 'win32a.inc'  
section '.data' data readable writeable  
 struct ticket  
 destination db 20 dup(0)  
 departure db 20 dup(0)  
 date db 20 dup(0)  
 cost dd 0  
 ends  
 tickets ticket ?,?,?,?  
 ticket ?,?,?,?  
 ticket ?,?,?,?  
 ticket ?,?,?,?  
 ticket ?,?,?,?  
 structOutput db '%s, %s, %s, %d',10,0  
 n dd 5  
 sizeOf dd 64  
 i dd 0  
 readFileName db 'in',0  
 writeFileName db 'out',0  
 wMode db 'w',0  
 rMode db 'r',0  
 readFile dd 0  
 writeFile dd 0  
 size dd 0  
section '.code' code readable writeable executable  
start:  
 invoke fopen, readFileName, rMode  
 ; сохранение дескриптора файла  
 mov [ds:readFile], eax

Продолжение листинга 1

; вывчислить количество имеющихся байт  
 mov eax, [ds:n]  
 imul [ds:sizeOf]  
 ; хотим выбрать 1 элемент  
 mov ecx, 1  
 invoke fread, tickets, ecx, eax, [ds:readFile]  
 invoke fclose, [ds:readFile]  
 invoke fopen, writeFileName, wMode  
 mov [ds:writeFile], eax  
 mov [ds:i], 0  
 loop1:  
 ; Расчет смещения к началу структуры  
 mov eax, [ds:i]  
 imul [ds:sizeOf]  
 mov edx, eax  
 add edx, tickets.cost  
 push eax  
 mov eax, 99  
 add eax, [edx]  
 mov [edx], eax  
 pop eax  
 inc [ds:i]  
 mov edx, [ds:i]  
 cmp edx, 5  
 jne loop1  
 ; расчет количества байт для записи в файл  
 mov eax, [ds:n]  
 imul [ds:sizeOf]  
 mov ecx, 1  
 mov [ds:i], 0  
 loop2:  
 ; расчет смещения к началу структуры  
 mov eax, [ds:i]  
 imul [ds:sizeOf]  
 mov [ds:size], eax  
 ; расчет полей  
 mov ebx, [ds:size]  
 add ebx, tickets.destination  
 mov ecx, [ds:size]  
 add ecx, tickets.departure  
 add eax, tickets.date  
 mov edx, [ds:size]  
 add edx, tickets.cost  
 invoke printf, structOutput, ebx, ecx, eax, [edx]  
 inc [ds:i]  
 mov edx, [ds:i]  
 cmp edx, 5  
 jne loop2  
 mov eax, [ds:n]  
 imul [ds:sizeOf]  
 mov ecx, 1  
 invoke fwrite, tickets, ecx, eax, [ds:writeFile]  
 invoke fclose, [ds:writeFile]  
 invoke getch  
 invoke ExitProcess, 0  
section '.idata' data import readable  
 library kernel, 'kernel32.dll',\  
 msvcrt, 'msvcrt.dll'

Окончание листинга 1

import kernel,\  
 ExitProcess, 'ExitProcess'  
 import msvcrt,\  
 printf, 'printf',\  
 getch, '\_getch', scanf, 'scanf', fopen, 'fopen', fwrite, 'fwrite', fclose,  
 'fclose', fread, 'fread'

Листинг 2 – Код программы для изначальной записи в файл

format PE Console  
entry start  
include 'win32a.inc'  
section '.data' data readable writeable  
 struct ticket  
 destination db 20 dup(0)  
 departure db 20 dup(0)  
 date db 20 dup(0)  
 cost dd 0  
 ends  
 tickets ticket 'krasnoyarsk','moscow','10.01.2024',100  
 ticket 'moscow','krasnoyarsk','11.01.2024',200  
 ticket 'spb','moscow','12.01.2024',250  
 ticket 'moscow','spb','13.01.2024',300  
 ticket 'spb','krasnoyarsk','14.01.2024',350  
 n dd 5  
 sizeOf dd 64  
 writeFileName db 'in',0  
 wMode db 'w',0  
 file\_d dd 0  
section '.code' code readable writeable executable  
start:  
 invoke fopen, writeFileName, wMode  
 mov [ds:file\_d], eax  
 mov eax, [ds:n]  
 imul [ds:sizeOf]  
 mov ecx, 1  
 invoke fwrite, tickets, ecx, eax, [ds:file\_d]  
 invoke fclose, [ds:file\_d]  
 invoke getch  
 invoke ExitProcess, 0  
section '.idata' data import readable  
 library kernel, 'kernel32.dll',\  
 msvcrt, 'msvcrt.dll'  
 import kernel,\  
 ExitProcess, 'ExitProcess'  
 import msvcrt,\  
 printf, 'printf',\  
 getch, '\_getch', scanf, 'scanf', fopen, 'fopen', fwrite, 'fwrite', fclose,  
 'fclose'

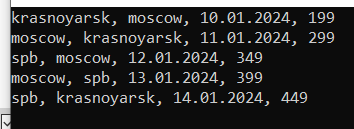


Рисунок 1 – Работа программы 1

**4 Вывод**

После изучения необходимой теории задание было выполнено.