Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Лабораторная работа № 9

Дисциплина: Машинно-зависимые языки программирования

Тема: Дизассемблирование

Студент: Платонова О. С.

Группа: ИУ7-45Б

Оценка (баллы) _____

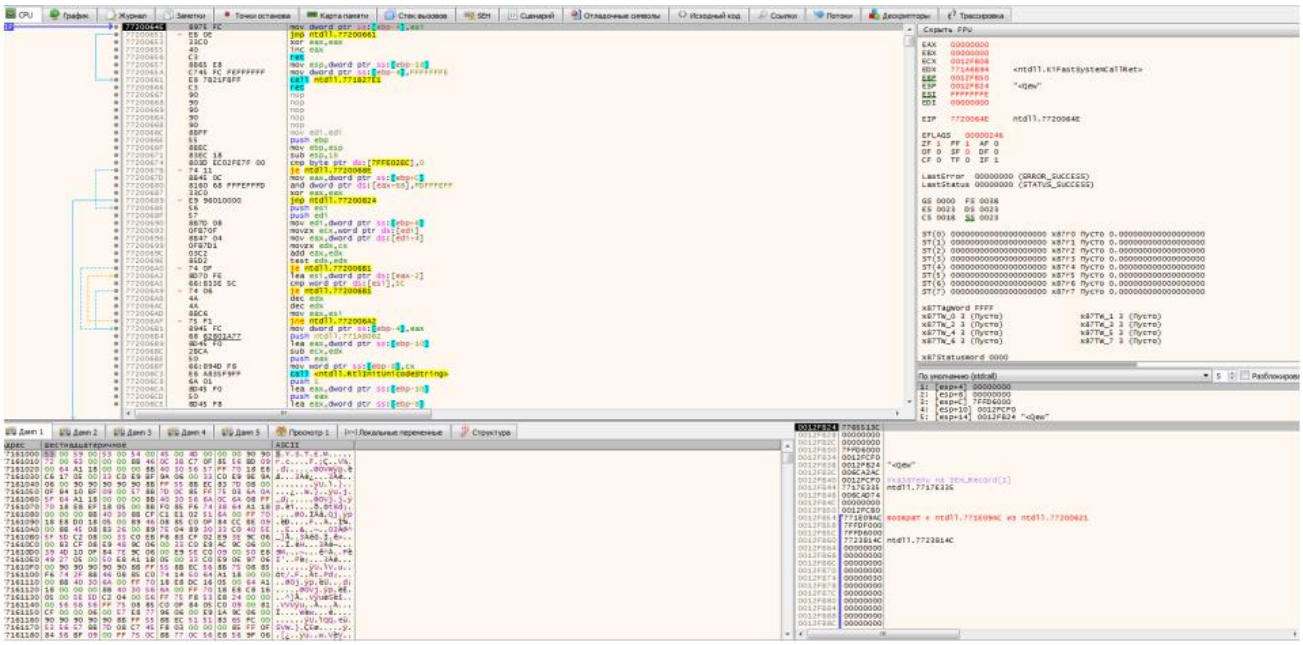
Преподаватель: Кузнецов Д. А.

Москва.
2020 г.

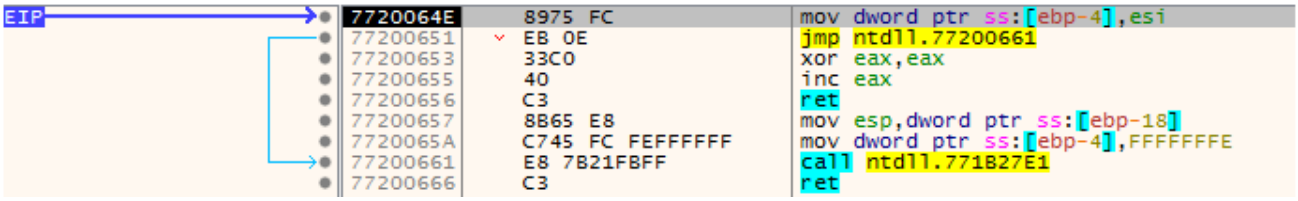
Выполним дизассемблирование файла BankCode.exe.

При запуске этого файла от пользователя требуется ввод кодового слова для восстановления пароля.

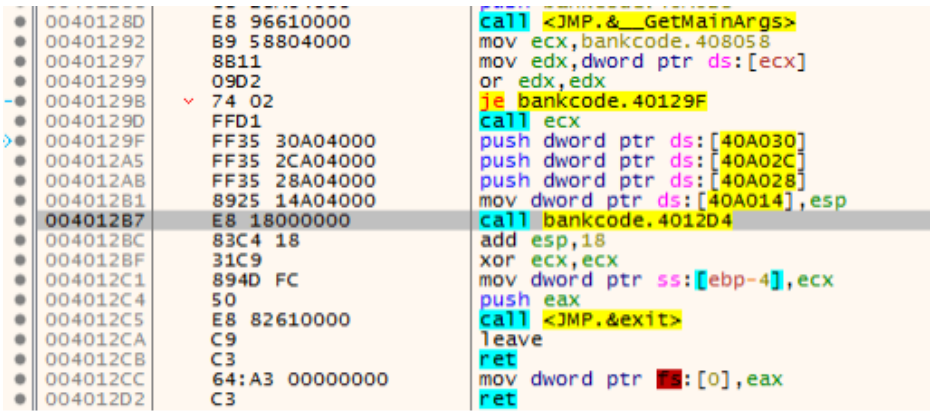
Откроем .exe файл в нашем отладчике. Вот так выглядит его интерфейс.



На данный момент мы оказались в точке входа.



Выполним трассировку «Анимация с обходом» и перейдем к дизассемблированному коду выделенной строки, – исследуемой процедуры.



Отметим, что перед вызовом процедуры в стеке размещается три указателя.

```
0040129F    FF35 30A04000       push dword ptr ds:[40A030]
004012A5    FF35 2CA04000       push dword ptr ds:[40A02C]
004012AB    FF35 28A04000       push dword ptr ds:[40A028]
004012B1    8925 14A04000       mov dword ptr ds:[40A014],esp
004012B7    E8 18000000         call bankcode.4012D4
```

Перейдя к дампу, легко увидеть, что это сообщения, предназначенные для пользователя.

```
0040A0A0 4B 65 79 20 77 6F 72 64 20 69 73 20 69 6E 63 6F Key word is inco
0040A0B0 72 72 65 63 74 0A 00 57 65 20 73 65 6E 74 20 6E rrect..We sent n
0040A0C0 65 77 20 70 61 73 73 77 6F 72 64 20 6F 6E 20 6D ew password on m
0040A0D0 61 69 6C 0A 00 09 49 6E 70 75 74 20 79 6F 75 72 ail...Input your
0040A0E0 20 63 6F 64 65 20 77 6F 72 64 0A 00 54 6F 20 72  code word..To r
0040A0F0 65 73 74 6F 72 65 20 70 61 73 73 77 6F 72 64 0A estore password.
```

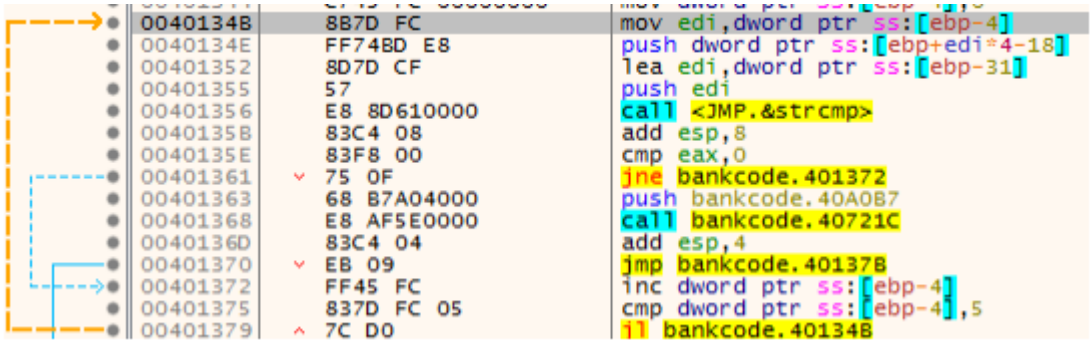
Это означает, что исследуемая функция ответсвенна за вызов функций ввода-вывода.

Рассмотрим процедуру.

Вот так, начиная с выделенной строки, выглядит наша процедура.

```
004012D4    55                  push ebp
004012D5    89E5                mov ebp,esp
004012D7    83EC 34             sub esp,34
004012DA    B9 0D000000         mov ecx,D
004012DF    49                  dec ecx
004012E0    C7048C 5A5AFAFF     mov dword ptr ss:[esp+ecx*4],FFFA5A5A
004012E7  ^ 75 F6               jne bankcode.4012DF
004012E9    57                  push edi
004012EA    8D3D 1EA14000       lea edi,dword ptr ds:[40A11E]
004012F0    897D E8             mov dword ptr ss:[ebp-18],edi
004012F3    8D3D 19A14000       lea edi,dword ptr ds:[40A119]
004012F9    897D EC             mov dword ptr ss:[ebp-14],edi
004012FC    8D3D 14A14000       lea edi,dword ptr ds:[40A114]
00401302    897D F0             mov dword ptr ss:[ebp-10],edi
00401305    8D3D 0AA14000       lea edi,dword ptr ds:[40A10A]
0040130B    897D F4             mov dword ptr ss:[ebp-C],edi
0040130E    8D3D 01A14000       lea edi,dword ptr ds:[40A101]
00401314    897D F8             mov dword ptr ss:[ebp-8],edi
00401317    68 ECA04000         push bankcode.40A0EC
0040131C    E8 FB5E0000         call bankcode.40721C
00401321    83C4 04             add esp,4
00401324    68 D5A04000         push bankcode.40A0D5
00401329    E8 EE5E0000         call bankcode.40721C
0040132E    83C4 04             add esp,4
00401331    8D7D CF             lea edi,dword ptr ss:[ebp-31]
00401334    57                  push edi
00401335    E8 36610000         call <JMP.&gets>
0040133A    83C4 04             add esp,4
0040133D    C745 FC 00000000    mov dword ptr ss:[ebp-4],0
00401344    C745 FC 00000000    mov dword ptr ss:[ebp-4],0
0040134B    8B7D FC             mov edi,dword ptr ss:[ebp-4]
0040134E    FF74BD E8           push dword ptr ss:[ebp+edi*4-18]
00401352    8D7D CF             lea edi,dword ptr ss:[ebp-31]
00401355    57                  push edi
00401356    E8 8D610000         call <JMP.&strcmp>
0040135B    83C4 08             add esp,8
0040135E    83F8 00             cmp eax,0
00401361  v 75 0F               jne bankcode.401372
00401363    68 B7A04000         push bankcode.40A0B7
00401368    E8 AF5E0000         call bankcode.40721C
0040136D    83C4 04             add esp,4
00401370  v EB 09               jmp bankcode.40137B
00401372    FF45 FC             inc dword ptr ss:[ebp-4]
00401375    837D FC 05          cmp dword ptr ss:[ebp-4],5
00401379  ^ 7C D0               jl bankcode.40134B
0040137B    837D FC 05          cmp dword ptr ss:[ebp-4],5
0040137F  v 75 0D               jne bankcode.40138E
00401381    68 A0A04000         push bankcode.40A0A0
00401386    E8 915E0000         call bankcode.40721C
0040138B    83C4 04             add esp,4
```

Заметим, что внутри процедуры выполняется сравнение строк в цикле (счетчик [ebp-4] увеличивается на 1).



В случае равенства строк (eax = 0) происходит ввод/вывод сообщения на экран.

```
call <JMP.&strcmp>
add esp,8
cmp eax,0
jne bankcode.401372
push bankcode.40A0B7
call bankcode.40721C
```

Перейдя к дампу, увидим, что это сообщение о корректном вводе.

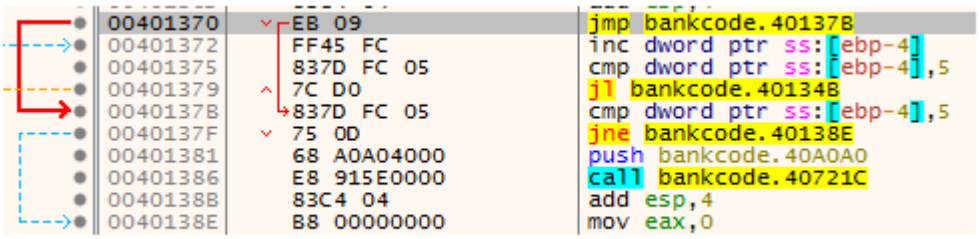
```
0040A0B7 57 65 20 73 65 6E 74 20 6E 65 77 20 70 61 73 73 We sent new pass
0040A0C7 77 6F 72 64 20 6F 6E 20 6D 61 69 6C 0A 00 09 49 word on mail...I
```

Значит, вся информация о доступе располагается в сегменте данных по адресу, начиная с [ebp – 8].

Действительно,

```
lea edi,dword ptr ds:[40A11E]   0040A11E:"Ivanov"
mov dword ptr ss:[ebp-18],edi
lea edi,dword ptr ds:[40A119]   0040A119:"1405"
mov dword ptr ss:[ebp-14],edi
lea edi,dword ptr ds:[40A114]   0040A114:"Mary"
mov dword ptr ss:[ebp-10],edi
lea edi,dword ptr ds:[40A10A]   0040A10A:"Childhood"
mov dword ptr ss:[ebp-C],edi
lea edi,dword ptr ds:[40A101]   0040A101:"Business"
mov dword ptr ss:[ebp-8],edi
```

В случае окончания цикла и неравенства введенной строки ни одной строке из базы данных, происходит вывод сообщения на экран и завершение процедуры.



Также рассмотрим функции считывания строки из консоли и сравнения строк.

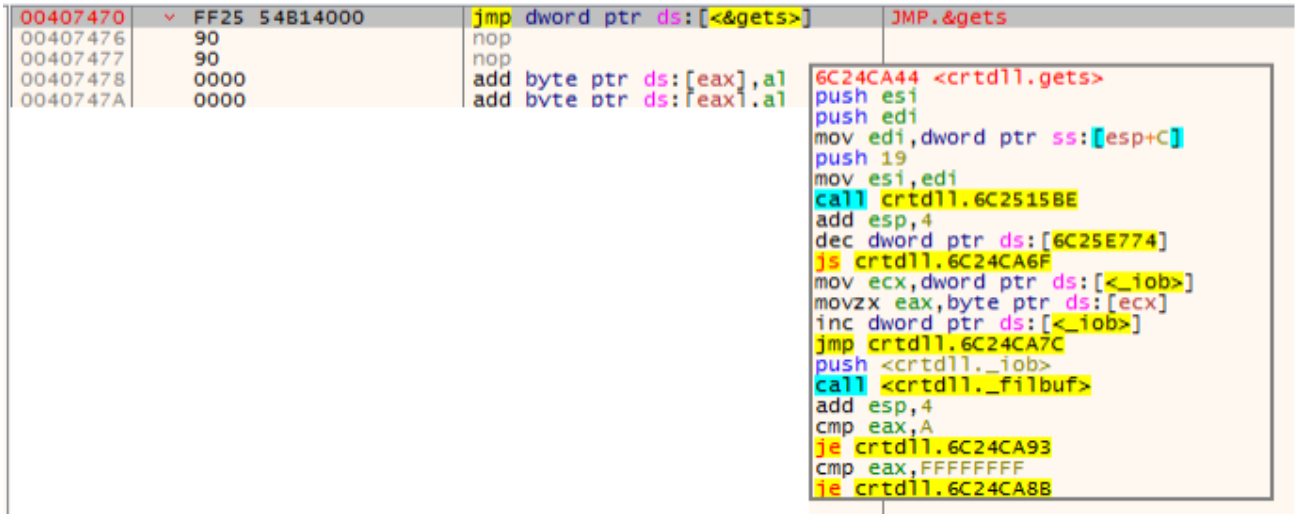
```
push edi
call <JMP.&gets>
```

```
push edi
call <JMP.&strcmp>
```

Функция сравнения строк

Функция использует команды обработки строк с префиксом повторения. И выполняет побитовое сравнение двух строк (ebp + 8; ebp + C) командой repe cmpsb. Предварительно вычисляется длина максимальной строки (командой repne scasb) и записывается в регистр ecx.

Функция считывания строки



Вывод

Мы научились проводить дизассемблирование exe файлов.