

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
“ЛЭТИ” ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра математического обеспечения и применения ЭВМ

ОТЧЕТ

По лабораторной работе №5

По дисциплине “Операционные системы”

Тема: Сопряжение стандартного и пользовательского обработчика прерываний

Студент гр. 8382

Гордиенко А.М.

Преподаватель

Ефремов М.А.

Санкт-Петербург

2020

Цель работы.

Исследование возможности встраивания пользовательского обработчика прерываний в стандартный обработчик от клавиатуры. Пользовательский обработчик прерывания получает управление по прерыванию (int 09h) при нажатии клавиши на клавиатуре. Он обрабатывает скан-код и осуществляет определенные действия, если скан-код совпадает с определенными кодами, которое он должен обрабатывать. Если скан-код не совпадает с этими кодами, то управление передается стандартному прерыванию.

Выполнение работы.

В процессе выполнения лабораторной работы был написан и отлажен программный модуль типа .exe, выполняющий следующие функции:

1. Проверяет, установлено ли пользовательское прерывание с вектором 09h.
2. Устанавливает резидентную функцию для обработки прерывания и настраивает вектор прерывания, если прерывание не установлено, и осуществляет выход по функции 4ch прерывания int 21h.
3. Если прерывание установлено, то выводится соответствующее сообщение и осуществляется выход по функции 4ch прерывания int 21h.
4. Выгрузка прерывания по значению командной строки /un.

Резидентная функция переназначает клавиши. Так, например, при нажатии клавиш d и g будет выводиться символ l, при нажатии f - будет выведен - o, при нажатии h - выведется символ пробела, при нажатии c, v, b – w, t, f соответственно.

Результат работы программы в различных состояниях показан на рисунке 1-4.

```

C:\>lr5.exe
Loaded
Tip: Try to print "dfghcvb"

C:\>lol wtf
Illegal command: lol.

C:\>lr5.exe
Already loaded

C:\>lr5.exe /un
Unloaded

C:\>lr5.exe /un
Nothing to unload

C:\>dfghcvb
Illegal command: dfghcvb.

```

Рисунок 1. Тестирование программы в различных состояниях.

```

C:\>LR3-1.COM
Available memory - 648912 B.
Extended memory - 15360 B.
MCB Type - 4D Sector - MS DOS Size -      16 Bytes. Last 8 bytes -
MCB Type - 4D Sector - FREE Size -       64 Bytes. Last 8 bytes -
MCB Type - 4D Sector - 0040 Size -      256 Bytes. Last 8 bytes -
MCB Type - 4D Sector - 0192 Size -      144 Bytes. Last 8 bytes -
MCB Type - 5A Sector - 0192 Size - 648912 Bytes. Last 8 bytes - LR3-1

```

Рисунок 2. Представление памяти до загрузки прерывания.

```

C:\>LR3-1.COM
Available memory - 647696 B.
Extended memory - 15360 B.
MCB Type - 4D Sector - MS DOS Size -      16 Bytes. Last 8 bytes -
MCB Type - 4D Sector - FREE Size -       64 Bytes. Last 8 bytes -
MCB Type - 4D Sector - 0040 Size -      256 Bytes. Last 8 bytes -
MCB Type - 4D Sector - 0192 Size -      144 Bytes. Last 8 bytes -
MCB Type - 4D Sector - 0192 Size -     1040 Bytes. Last 8 bytes - LR5
MCB Type - 4D Sector - 01DE Size -     1144 Bytes. Last 8 bytes -
MCB Type - 5A Sector - 01DE Size - 647696 Bytes. Last 8 bytes - LR3-1

```

Рисунок 3. Представление памяти после загрузки прерывания.

```

C:\>LR3-1.COM
Available memory - 648912 B.
Extended memory - 15360 B.
MCB Type - 4D Sector - MS DOS Size -      16 Bytes. Last 8 bytes -
MCB Type - 4D Sector - FREE Size -       64 Bytes. Last 8 bytes -
MCB Type - 4D Sector - 0040 Size -      256 Bytes. Last 8 bytes -
MCB Type - 4D Sector - 0192 Size -      144 Bytes. Last 8 bytes -
MCB Type - 5A Sector - 0192 Size - 648912 Bytes. Last 8 bytes - LR3-1

```

Рисунок 4. Представление памяти после выгрузки прерывания.

Контрольные вопросы.

1. Какого типа прерывания использовались в работе?

Функция 09h для доступа к клавиатуре. Также сами программные прерывания int 21h, и int 16h.

2. Чем отличается скан-код от кода ASCII?

Скан-код - код-значение, присвоенное сигналу нажатия соответствующей клавиши. Сигнал не может изменяться от переназначения символа. Так значения клавиш упорядочены по возрастанию слева направо, сверху вниз от клавиши Esc. Значение ASCII же это некоторая упорядоченная система нумерации символов. Так, например, символ 'F' и 'f' будут иметь разные значения, в то время как скан-коды клавиатуры будут одинаковыми.

Выводы.

В ходе работы был реализован пользовательский обработчик от клавиатуры.