**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 10**

**Студентки гр. И-2-15(а)**

**Авдиль Салие**

**По дисциплине**: Разработка ПО для встроенных систем.

**Тема занятия**: Поиск информации в оперативной памяти

**Цель занятия**: применение алгоритмов поиска информации в различных сегментах оперативной памяти

**Задание:** необходимо получить из среды окружения строки, содержащие заданный символ. Если строки/строк с заданным символом не существует – выдать сообщение «Строк с заданным символом не существует».

**Листинг файла lab10.asm**

**.model small**; *Указание модели памяти: указывает, что сегмент данных (он начинается с директивы.DATA) и сегмент кода (начинается с директивы.CODE) будут занимать не более одного сегмента ОЗУ ( 64 Кбайт).*

**.286**; *разрешены команды 286-ого микропроцессора*

**.data**; *директива описывает обычный сегмент данных (переменных).*

**a db 'Stroki s takim simvolom net!$'**

**c db ";"**

**w db 10,13,'$'**

**.code**; *объявление сегмента кода*

*; Метка входа в программу*

**start: mov ax, @data**; *пересылаем адрес сегмента данных в регистр* ax*,*

**mov ds, ax** ; *устанавливаем регистр* ds *на сегмент данных*

**mov dx, offset w**; *смещение переменной* w *в регистр* dx

**mov ah, 9h**; *9-ая функция 21-го прерывания для вывода*

**int 21h**; *строки на экран*

**mov bl, c**

**mov ax, [es:2ch]**; *пересылаем в регистр ax адрес среды окружения,*

**mov ds, ax**; *расположенный по смещению 2ch в PSP*

**mov di, 0**; *зануляем* di *– текущий символ.*

**mov si, 0**; *зануляем* si – *смещение в среде окружения*

**mov bh, 0**; *зануляем* bh – *флаг для вывода строки*

**k4: mov dl, [ds:di]**; *пересылаем в регистр* dl *символ среды окружения*

**inc di**; *увеличиваем* di *на единицу*

**cmp dl, bl**; *проверка символа на равенство с символом*

**je k3**; *если нашли искомый символ переходим в* k3

**k1: cmp dl, 0**; *проверка конца строки*

**jne k4**; *если не равно переходим в* k4

**mov dl, [ds:di]**; *считать следующий символ за* 0

**cmp dl, 0**; *проверка конца среды окружения*

**je k2**; *если конец переходим в* k2

**mov si, di**; *если это начало новой строки, обновляем индекс*

**jmp k4**; *переходим в* k4

**k3: mov bh, 1**; *если строка найдена, меняем флаг*

**mov dl, 0**;

**mov dl, [ds:si];**

**cmp dl, 0;**

**je k8;**

**mov di, si;**

**mov ah, 2h**; *02h-номер функции DOS, которая выводит символ*

**int 21h**; *в STDOUT вывод содержимого регистра dl*

**inc si**; *увеличиваем* si *на единицу*

**jmp k3***; переходим в* k3

**k8: mov di, si;**

**mov ax, @data;**

**mov ds, ax;**

**mov dx, offset w**; *смещение переменной w в регистр dx*

**mov ah, 9h**; *9-ая функция 21-го прерывания для вывода*

**int 21h**; *строки на экран*

**mov ax, [es:2ch]**; ; *пересылаем в регистр* ax *адрес среды окружения,*

**mov ds, ax**; *расположенный по смещению 2ch в PSP*

**jmp k4**;*переходим в* k4

**k2: cmp bh, 1**; *если символ не найден, вывести строку об этом*

**je k5**; *если найдена переходим в* k5

**mov ax, @data;**

**mov ds, ax;**

**mov dx, offset a**; *вывод строки*

**mov ah, 9h;**

**int 21h;**

**k5: mov ah,4ch**; *Завершение программы*

**int 21h;**

**end start;**

**Результат выполнения файла lab10.asm**

