

Ильченко Ева ИУ7-24Б

Отчет по Заданию №3.3. Представление в памяти строк и массивов строк

Цель: изучить на практике, как в памяти компьютера представлены строки и массивы строк

Строка

Строка

```
char s[] = "devochka-koshochka"
```

Дамп памяти

```
(gdb) x /19xb arr1
0x7ff7bfeff430: 0x64 0x65 0x76 0x6f 0x63 0x68 0x6b 0x61
0x7ff7bfeff438: 0x2d 0x6b 0x6f 0x73 0x68 0x6f 0x63 0x68
0x7ff7bfeff440: 0x6b 0x61 0x00
```

Каждый байт отвечает за определенную букву, так как одинаковые буквы имеют одинаковые коды. Также можно заметить, что байты соотносятся с кодом буквы в таблице `ascii`. Заканчивается строка терминальным нулем с кодом 0

Массив строк

1. Массив строк как матрица

Строка

```
char arr1[][16] = {"Frank Ocean", "Kosmonavtov net", "The Weeknd"};
```

Дамп всей матрицы

```
(gdb) x /48xb arr1
0x7ff7bfeff420: 0x46 0x72 0x61 0x6e 0x6b 0x20 0x4f 0x63
0x7ff7bfeff428: 0x65 0x61 0x6e 0x00 0x00 0x00 0x00 0x00
0x7ff7bfeff430: 0x4b 0x6f 0x73 0x6d 0x6f 0x6e 0x61 0x76
0x7ff7bfeff438: 0x74 0x6f 0x76 0x20 0x6e 0x65 0x74 0x00
0x7ff7bfeff440: 0x54 0x68 0x65 0x20 0x57 0x65 0x65 0x6b
0x7ff7bfeff448: 0x6e 0x64 0x00 0x00 0x00 0x00 0x00 0x00
```

На каждую строку в массиве `arr1` выделяется 16 байт памяти. Поэтому сначала записываются коды букв в память, а если длина строки меньше 16 символов, то записываются терминальные нули в свободные ячейки.

Всего 3 строки, каждая занимает 16 байт, следовательно суммарный размер памяти, который занимает массив равен $16 * 3 = 48$

При этом размер “вспомогательных” данных равен: суммарный размер массива - сумма длин строк с терминальными нулями.

В этом данном примере:

размер “полезных” данных: 39 байт

размер “вспомогательных” данных: 9 байт

% «вспомогательных» данных по отношению ко всем данным: 18,75 %

2. Массив строк как массив указателей

Строка

```
char *arr2[] = {"Chipsiki", "Coca-cola", "Morozhenko",  
"Marmeladki"};
```

Дамп массива указателей

```
(gdb) x /32xb arr2  
0x7ff7bfeff400: 0x87 0x3f 0x00 0x00 0x01 0x00 0x00 0x00  
0x7ff7bfeff408: 0x90 0x3f 0x00 0x00 0x01 0x00 0x00 0x00  
0x7ff7bfeff410: 0x9a 0x3f 0x00 0x00 0x01 0x00 0x00 0x00  
0x7ff7bfeff418: 0xa5 0x3f 0x00 0x00 0x01 0x00 0x00 0x00
```

Дамп строки 1

```
(gdb) x /9xb *arr2  
0x100003f87: 0x43 0x68 0x69 0x70 0x73 0x69 0x6b 0x69  
0x100003f8f: 0x00
```

Дамп строки 2

```
(gdb) x /10xb *(arr2+1)  
0x100003f90: 0x43 0x6f 0x63 0x61 0x2d 0x63 0x6f 0x6c  
0x100003f98: 0x61 0x00
```

Дамп строки 3

```
(gdb) x /11xb *(arr2+2)  
0x100003f9a: 0x4d 0x6f 0x72 0x6f 0x7a 0x68 0x65 0x6e  
0x100003fa2: 0x6b 0x6f 0x00
```

Дамп строки 4

```
(gdb) x /11xb *(arr2+3)  
0x100003fa5: 0x4d 0x61 0x72 0x6d 0x65 0x6c 0x61 0x64  
0x100003fad: 0x6b 0x69 0x00
```

В `arr2` хранятся указатели на строки, каждый указатель занимает 8 байт памяти. Под каждый указателем хранится строка, которая занимает ровно отведенное для нее пространство. Строки располагаются друг за другом в памяти

Таким образом размер “полезных” данных - вся занимаемая массивом память