

Ильченко Ева ИУ7-24Б

Отчет по Заданию №3.4. Представление в памяти структур и объединений

Цель: изучить на практике, как в памяти компьютера представлены структуры и объединения

Локальные переменные

Описание локальных переменных:

```
short int a = 10;  
int b = 100;  
long int c = 1000;  
char d = 'a';  
float e = 3.14;  
double f = 3.1416;  
unsigned int g = 5;
```

Дамп памяти со всеми локальными переменными

```
(gdb) x /2xb &a  
0x7ff7bfeff44e: 0x0a 0x00  
(gdb) x /4xb &b  
0x7ff7bfeff448: 0x64 0x00 0x00 0x00  
(gdb) x /8xb &c  
0x7ff7bfeff440: 0xe8 0x03 0x00 0x00 0x00 0x00 0x00 0x00  
(gdb) x /1xb &d  
0x7ff7bfeff43f: 0x61  
(gdb) x /4xb &e  
0x7ff7bfeff438: 0xc3 0xf5 0x48 0x40  
(gdb) x /8xb &f  
0x7ff7bfeff430: 0xa7 0xe8 0x48 0x2e 0xff 0x21 0x09 0x40  
(gdb) x /4xb &g  
0x7ff7bfeff42c: 0x05 0x00 0x00 0x00
```

Сведения о переменных

Имя	Размер	Адрес
a	2	0x7ff7bfeff44e
b	4	0x7ff7bfeff448
c	8	0x7ff7bfeff440
d	1	0x7ff7bfeff43f
e	4	0x7ff7bfeff438
f	8	0x7ff7bfeff430

g	4	0x7ff7bfeff42c
---	---	----------------

Можно заметить, что переменные располагаются друг за другом в памяти, занимая место, равное размеру этой переменной

Структуры

Описание структуры и структурной переменной

```
struct song_t {
    char first_letter;
    int release_date;
    double duration;
};
struct song_t song = {'a', 2023, 2.43};
```

```
sizeof(song) == 13
```

Дамп памяти структурной переменной

```
(gdb) x /16xb &song
0x7ff7bfeff440: 0x61 0x9c 0x00 0x00 0xe7 0x07 0x00 0x00
0x7ff7bfeff448: 0x71 0x3d 0x0a 0xd7 0xa3 0x70 0x03 0x40
```

Сведения о полях структурной переменной

Имя	Размер	Адрес
first_letter	1	0x7ff7bfeff440
release_date	4	0x7ff7bfeff444
duration	8	0x7ff7bfeff448

Переменные располагаются друг за другом в памяти, занимая место, равное их размеру с учетом выравнивания

Переменная структурного типа располагается по адресу: 0x7ff7bfeff440 - адрес ее первой переменной. На это повлиял тип char

Описание упакованной структуры

```
#pragma pack(push, 1)
struct song_t {
    char first_letter;
    int release_date;
    double duration;
};
#pragma pack(pop)

struct song_t song = {'a', 2023, 2.43};
```

```
sizeof(song) == 13
```

Дамп памяти упакованной структурной переменной

```
(gdb) x /13xb &song
0x7ff7bfeff443: 0x61 0xe7 0x07 0x00 0x00 0x71 0x3d 0x0a
0x7ff7bfeff44b: 0xd7 0xa3 0x70 0x03 0x40
```

Сведения о полях упакованной структурной переменной

Имя	Размер	Адрес
first_letter	1	0x7ff7bfeff443
release_date	4	0x7ff7bfeff444
duration	8	0x7ff7bfeff448

Переменные располагаются друг за другом в памяти, занимая количество памяти точно равное их размеру

Переменная структурного типа располагается по адресу: 0x7ff7bfeff443 - адрес ее первой переменной

При расположении

```
struct song_t {
    double duration;
    int release_date;
    char first_letter;
};
```

структура занимает наименьшее количество места

Присутствует “завершающее” выравнивание. Оно равно 3 байтам

Объединения

Описание объединения и инициализация одного из его полей

```
union u_t{
    double a;
    int b;
    char c;
};

union u_t u = {100};
```

Дамп памяти объединения (поле a имеет значение 100)

```
0x7ff7bfeff448: 0x00 0x00 0x00 0x00 0x00 0x00 0x59 0x40
```

Дамп памяти объединения (поле b имеет значение 100)

```
0x7ff7bfeff448: 0x64 0x00 0x00 0x00 0x00 0x00 0x00 0x00
```