

Семинар #8: Указатели и динамическое выделение памяти.

Домашнее задание.

Память

Задача #1: Как выглядит память, инициализируемая при создании следующих переменных (в системе с порядком байт Little Endian):

```
1. int a = 0x11223344;
2. int b = 123456789;
3. int array[3] = {10, 2000, 65535};
4. char str[8] = "Hello";
5. float x = -15.91;
6. double y = -15.91;
7. struct data
{
    char str[5];
    int number;
};
struct data c = {"Cat", 100000};
```

Память представить в виде последовательности 2-значных шестнадцатиричных чисел. Например число `int a = 757004;` будет храниться в памяти как `0x54, 0x82, 0x73, 0x00`.

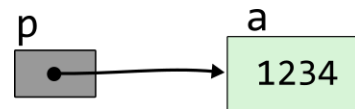
Подсказка: Чтобы проверить, как будет выглядеть память, можно создать указатель типа `char*` на эту память и распечатать каждый байт в виде шестнадцатиричного числа:

```
char* p = (char*)&a;
for (int i = 0; i < sizeof(a); ++i)
{
    printf("0x%02hhx ", p[i]);
}
```

Указатели

Указатель на int

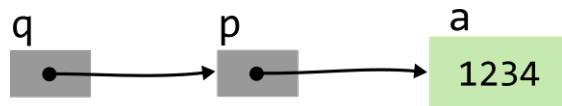
```
int a = 1234;
int* p = &a;
```



Задача #2: Удвойте значение переменной `a`, используя только указатель `p`.

Указатель на указатель на int

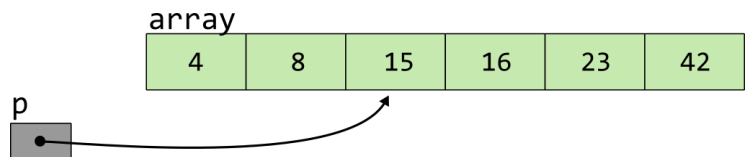
```
int a = 1234;
int* p = &a;
int** q = &p;
```



Задача #3: Удвойте значение переменной `a`, используя только указатель `q`.

Указатель на элемент массива

```
int main()
{
    int array[6] = {4, 8, 15, 16, 23, 42};
    int* p = &array[2];
}
```

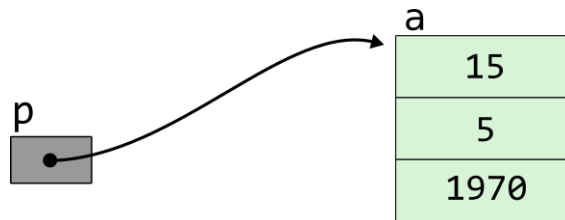


Задача #4: Удвойте значение `array[1]`, используя только указатель `p`.

Указатель на структуру

```
#include <stdio.h>
struct date
{
    int day;
    int month;
    int year;
};

int main()
{
    struct date a = {15, 5, 1970};
    struct date* p = &a;
    printf("%d %d %d\n", a.day, (*p).day, p->day);
}
```



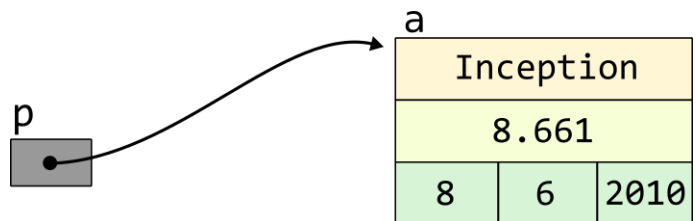
Задача #5: Удвойте значение поля `year`, используя только указатель `p`.

Указатель на структуру немного сложнее

```
#include <stdio.h>
struct movie
{
    char title[50];
    float rating;
    struct date release_date;
};

typedef struct movie Movie;

int main()
{
    Movie a = {"Inception", 8.661, {8, 6, 2010}};
    Movie* p = &a;
}
```



Задача #6: Удвойте значение поля `rating`, используя только указатель `p`.

Задача #7: Удвойте значение поля месяца выхода фильма, используя только указатель `p`.

Указатель на массив структур

```
#include <stdio.h>
struct movie
{
    char title[50];
    float rating;
    struct date release_date;
};
typedef struct movie Movie;

int main()
{
    Movie a[3] = {"Inception", 8.661, {8, 6, 2010}},
               {"Green Mile", 9.062, {6, 12, 1999}},
               {"Leon", 8.679, {14, 9, 1994}};
    Movie* p = &a[1];
}
```

p

A small grey square box with a black dot inside, labeled 'p' above it. An arrow points from the dot to the 'Green Mile' row of the array 'a'.

a		
"Inception"		
8.661		
8	6	2010
"Green Mile"		
9.062		
6	12	1999
"Leon"		
8.679		
14	9	1994

Задача #8: Удвойте значение рейтинга фильма Inception, используя только указатель p.

Задача #9: Удвойте значение года выхода фильма Leon, используя только указатель p.

Memory allocation

Основные функции для динамического выделения памяти:

- `void* malloc(size_t n)` – выделяет `n` байт и возвращает указатель `void*` на начало этой памяти
- `void free(void* p)` – освобождает выделенную память
- `void* realloc(void* p, size_t new_n)` – перевыделяет выделенную память

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int number_of_elements = 20;
    int* p = (int*)malloc(number_of_elements * sizeof(int));

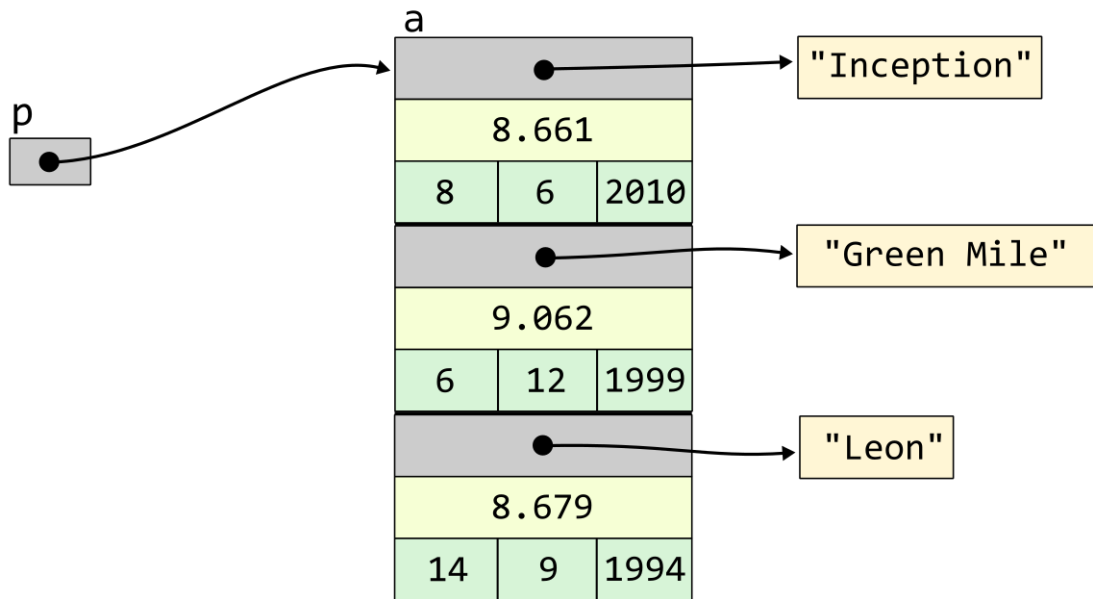
    free(p);
}
```

Указатель на массив структур, выделенный в куче

```
#include <stdio.h>
struct movie
{
    char title[50];
    float rating;
    struct date release_date;
};
typedef struct movie Movie;

void set_movie(Movie* pm, char* title, float rating, int day, int month, int year)
{
    pm->title = (char*)malloc(strlen(title) + 1);
    strcpy(pm->title, title);
    pm->rating = rating;
    pm->release_date.day = day;
    pm->release_date.month = month;
    pm->release_date.year = year;
}

int main()
{
    Movie* p = (Movie*)malloc(3 * sizeof(Movie));
    set_movie(p, "Inception", 8.661, 8, 6, 2010);
    set_movie(p + 1, "Green Mile", 9.062, 6, 12, 1999);
    set_movie(p + 2, "Leon", 8.679, 14, 9, 1994);
}
```



Стек

Куча

