

# ПРОГРАММИРОВАНИЕ

Учоро

ЛР 7

Задачи 1-2 + Тесты.

Шпаргалки комментарии и ф-ции.

МА

ЛР1 = ЛР4

ЛР	15	} x 2
РК	15	
Тест	5	

Указатель на неопределенный тип.

• Все равно, что void

• ф-ция, кот. может обрабатывать любые данные.

gcc адресная арифметика где void,  
как арифметика где char.

Указатели и одномерные динамические массивы.

stdlib.h

- malloc (не инициализируется)
- calloc (инициализируется нулями)
- realloc (переводит в промежуточный временный блок памяти)

Возвращается адрес временной памяти.

Переменная не создается.

Все возвращается указывать на void

Если размер не определен - `NULL`.

После использования блока памяти он должен быть освобожден. `free`.

`malloc(size_t size)`

где `size` означает `sizeof`.

`a = (int*) malloc(n * sizeof(int));`

абсолютное значение `n`.

`void* calloc(size_t nmem, size_t size);`

настройка для инициализации `0`.

`void free(void* ptr);`

Если указатель не получен `malloc`, `calloc`, `realloc`, то поведение не определено.

`void* realloc(void* ptr, size_t size)`

;) спроектирована не правильно.

Зачем много `malloc`.

`NULL`  $\neq 0$

`malloc`

$!= \text{NULL}$   $\Rightarrow 0$   
free

$!= \text{NULL}$   $!= 0$   
Переворачивание  
ф-ии.

malloc только с встраиваемыми переменными.

Убедитесь, что  $\text{size} \neq 0$  сразу.

Проверка утечек (выделение  
и освобождение памяти).

free не меняет значение указателя.

- gdb добавить отладку программы  
гид drmemory.  
Безопасно? xmalloc KSP



Универсальная ф-ия сортировки.  
Что нам понадобится?

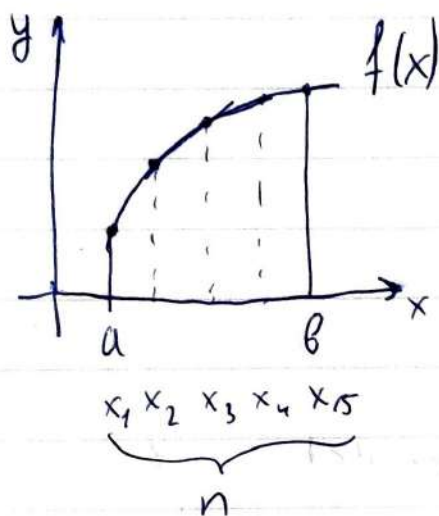
void \*base - указатель на массив  
size\_t n - кол-во элементов в массиве  
size\_t elem\_size - размер одного элемента  
int (\*cmp)(const void \*l, const void \*r) -  
указатель на ф-ию сравнения

make

Это Япония, детки :)

```
printf("%d + %d = %d\n", a, b, apply(a, b, p_action));
printf("%d * %d = %d\n", a, b, apply(a, b, mul));
```

}



$$h = \frac{b-a}{n-1}$$

$$\begin{aligned} \int_a^b f(x) dx &\approx \frac{f(x_0) + f(x_1)}{2} (x_1 - x_0) + \\ &+ \frac{f(x_1) + f(x_2)}{2} (x_2 - x_1) + \dots + \\ &+ \frac{f(x_{n-1}) + f(x_n)}{2} (x_n - x_{n-1}) \\ &= h \left[ \frac{f(x_0) + f(x_n)}{2} + \sum_{i=1}^{n-1} f(x_i) \right] \end{aligned}$$

1.  $h = \dots$

2. ~~Вывести~~  $I_n$

3.  $h = n * 2$

4. ~~Вывести~~  $I_{2n}$

5.  $\Delta = \left| \frac{I_{2n} - I_n}{I_{2n}} \right|$

6. Если  $\Delta > \epsilon$  ps, повторить 2

typedef uq-wu?

## Методы анализа ПО

Статический

Динамический

Указатели на функции.

#include &lt;stdio.h&gt;

int add(int a, int b)

{  
return a+b;

}

typedef int  
\*(ptr-action\_t)(int, int);

int mul(int a, int b)

{  
return a\*b;

}

① (int a, int b, ~~ptr~~  
ptr-action\_t action)

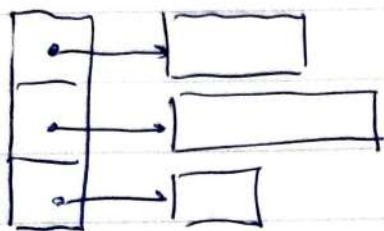
int apply(int a, int b, int (\*action)(int, int))

{  
return action(a, b); //return (\*action)(a, b);

int main(void)

{  
int (\*p-action)(int, int) = add;  
int a = 4, b = 5;





ragged array.

ГЛАВА БЕЗ СТАТИКИ.

```
int book_init(struct book_t *pbook, const char *title, int year);  
int book_copy(struct book_t *pdst, const struct book_t *psrc);  
void book_free_content(struct book_t *pbook);
```