

Konzeptuelle Aspekte

1. Typen geben

int int a = 51

char char a = 'x'

bool bool a = true

short short a = -10

long long a = -100

float float a = -10.45

double double b = 0.00105

2. класс парком: code - парком
load:

stack - блок парком, создается
для каждого парком, для каждого
комм. парком вводится.

heap - функция введена парком
Heap парком - парком с парком
введенные парком.

3. $\text{int } x$ $\text{int } xy = \&x$

~~int~~ от load парком можно
взять адрес при парком
введен адреса. & - парком введен
указатель.

$\text{int } z = y$ - указатель парком
парком введен парком на
конкретный указатель

4. К узлам можно прибавлять
уже есть, и менять свои
фрагменты.

8. Если - это перемещение, то
в листе; если - это и другие
фрагменты.

Если - все перемещение в лист
и не имеет своей цели.

Memory layout
int int = 90;

{

int * int_ptr = ∫ ^{int} указатель

(* int_ptr)++;

int * heap_ptr = new int;

* heap_ptr = 777. указатель

с 777 оценок
о перемещении

}

6. Функции

вызовов. `void(типично)`

{

внутренний код.

}

объявление

`void(перец1, ..., перецN) // void`

возвращает `тип void = void(...)`

7 static суммарно прописаны
имена переменных и
стандарт, при этом сохраняются
зависимости. После ввода из файла код
где переименованы

8 Возврат из функции

по значению

по адресу

по ссылке

9. CS & dio blog - fenv,

scanf("...", blog)

printf("...", blog)

10. blog - bblog istream, ostream

std::cin - blog

std::cout << bblog << std::endl

11) part 1

std::ofstream wtf pma - zanyu

wtf << zanyu << std::endl

std::setw()

setfill() neg. zanyu

wtf.close() - zanyu zanyu

~~std::~~

file = fopen(wtf) -