Семинар 2 – преамбула

МГТУ им. Н.Э. Баумана

February 26, 2016

Ввод и вывод

C:

```c
#include <stdio.h>
// ...
char c = 'a';
int n = 10;
double d = 30.;
// ...
scanf("%c", &c);
scanf("%d", &n);
// ...
printf("%c %d %lf\n", c, n, d);
//
```

C++:

```cpp
#include <iostream>
// ...
char c = 'a';
int n = 10;
double d = 30.;
// ...
std::cin >> c;
std::cin >> n;
// ...
std::cout << c << " " << n << " "
      << d << std::endl;
```

Ввод строк

```cpp
#include <string>
// ...
std::string s;
char cstr[256];
// ...
std::cin >> s; // read till space
std::cin.getline(cstr, sizeof(cstr)); // till line end
// ...
std::cout << s << " " << cstr << std::endl;
```

Выделение динамической памяти

Указатель на одно значение

C:

```c
int* a=malloc(sizeof(int));
*a=0
// ...
free(a);
```

C++:

```c
int* a=new int;
// ...
delete a;
```

Указатель на массив

C:

```c
int* a=malloc(10*sizeof(int));
// ...
free(a);
```

C++:

```c
int* a=new int[10];
// ...
delete[] a;
```

new позволяет звать конструктор объекта

```c
int* a=new int(0); // *a is initialized
int* b=new int(*a); // *b is a copy of *a
// ...
// T* t = new T(param1, param2); // any ctor appropriate
delete a;
delete b;
```

Перегрузка функций

Нет ошибок компиляции:

```
1 bool ge(int i1, int i2) { return i1 >= i2; }
2 bool ge(double d1, int d2) { // same name
3   // params type differs
4   return d1 >= d2;
5 }
6 bool ge(double d1, int d2, int upper_bound) { // same name
7   // params count differs
8   return d1 >= d2;
9 }
```

Ошибка компиляции:

```
1 bool ge(int i1, int i2) {
2   return i1 >= i2;
3 }
4 char ge(int i1, int i2) { return i1 >= i2; } // ERROR: return type
      difference is not enough
5 bool ge(int i1, int i2, int ub = 10) { return i1 >= i2; } // ERROR: can
      be used with default param
```

Самостоятельно:

Поиск/выбор перегруженной функции.

Использование ссылок

Ссылка (lvalue-ссылка)

- как указатель
- не требуется * при доступе
- ссылка не может указывать на другой объект после инициализации
- не может указывать на rvalue, если не const

```
// C:
int a = 0;
int *b = &a; //pointer
*b = 1
assert(a == 1);
```

```
// C++:
int a = 0;
int& b = a; // reference
b = 1
assert(a == 1);
```

```
int f(const int& a) { return a+10; }
int g(int& a) { return (++a)+10; }
// ...
int b = 5;
f(b); // ok
f(3); // ok -- const reference
g(b); // ok, b == 6
g(8); // ERROR: 8 is rvalue
```

Range-based for

C:

```
1 int a[10] ={0};
2 for (int i = 0; i < sizeof(a)/sizeof(a[0]); ++i)
3   a[i] += 1;
```

C++:

```
1 int a[10] ={0};
2 for (int& val : a) // Note the reference
3   v += 1;
```

Ввод и вывод в файлы

C:

```c
#include <stdio.h>
// ...
char c = 'a';
int n = 10;
double d = 30.;

FILE* fp = fopen("1.txt", "w");
fprintf(fp, "%c %d %lf\n", c, n,
    d);
fclose(fp);


fp = fopen("1.txt", "r");
fscanf(fp, "%c", &c);
fscanf(fp, "%d", &n);
fclose(fp)
//
```

C++:

```cpp
#include <fstream>
// ...
char c = 'a';
int n = 10;
double d = 30.;
{
  std::fstream fout("1.txt", std
      ::fstream::out);
  fout << c << " " << n << " " <<
       d << std::endl;
} //closed on destruction
{
  std::fstream fin("1.txt", std::
      fstream::in);
  fin >> c;
  fin >> n;
} //closed on destruction
```

TODOs

TODO:

- Вывести "Hello, world"
- Получить введенное пользователем число N
- Вывести на двух строчках (числа от 1 до N) и (числа от N до 1)

TODO:

- Реализовать функцию swap для целых чисел без указателей
- Реализовать функцию swap для вещественных чисел без указателей
- Проверить, что для каждого типа зовется своя функция

TODO:

- Получить введенное пользователем число N
- Создать и заполнить числами целочисленный массив размера N
- Создать целочисленный массив и записать в него квадраты элементов первого массива
- Вывести второй массив

TODOs

TODO:

- Создать массив со случайными целыми числами размера, указанного пользователем
- Записать массив в файл.
- Считать массив из файла.
- Для всех элементов массива выполнить преобразование x**31 mod 1023, результат записать в новый массив.
- Отсортировать оба массива.
- Выполнить слияние массивов в один с сохранением упорядоченности
- Записать результат в файл
- Почистить память

Промежуточные данные о выполнении действий выводите в поток вывода