

Отчет о домашнем задании по «Архитектуре вычислительных систем» #1

Студент: Кунин Илья, БПИ203

1. Вариант задания

Variant number = 96

Number of task = 12

Number of function = 7

12. Животные	1. Рыбы (место проживания – перечислимый тип: река, море, озеро...) 2. Птицы (отношение к перелету: перелетные, остающиеся на зимовку – булевская величина) 3. Звери (хищники, травоядные, насекомоядные... – перечислимый тип)	1. Название – строка символов, 2. Вес в граммах (целое)	Частное от деления суммы кодов незашифрованной строки на число символов в этой строке на вес (действительное число)
--------------	---	--	---

7. Упорядочить элементы контейнера по возрастанию используя сортировку методом деления пополам (Binary Insertion). В качестве ключей для сортировки и других действий используются результаты функции, общей для всех альтернатив.

2. Примеры команд запуска (инструкций)

Команды необходимо запускать из папки с исполняемым файлом

```
./task -n 200 files/output1.txt files/output2.txt
```

```
./task -f files/input.txt files/output1.txt files/output2.txt
```

3. Сортировка

В программе производится Binary Insertion Sort по возрастанию параметра Value.

4. Как должен быть устроен файл INPUT.TXT (Формат данных)

Каждая строка состоит из 4 слов и идет в формате A B C D, где

- ❖ A - число от 0 до 2, 0 - FISH, 1 - BIRD, 2 – BEAST
- ❖ B - имя (строка без пробелов)
- ❖ C - вес (целое число в граммах)
- ❖ D - специальная характеристика. FISH (0-3). BIRD (0,1). BEAST (0-2).

Пример строки файла: 0 name 100 1

5. Содержание архива

Файл task – скомпилированный, с помощью CMake, файл программы

Файл help.txt – небольшой файл с сохраненными командами

Файл отчет.pdf - этот файл

Подкаталог code – исходные тексты программы

Подкаталог build – папка с техническими файлами «билда» программы

Подкаталог files – папка для хранения файлов вывода и ввода

Подкаталог tests – папка с двумя подкаталогами – тестовыми входными наборами и результатами работы на этих наборах (выходные файлы)

6. Основные характеристики программы

Название входного теста – количество объектов – время работы (сек.)

Число внешних интерфейсных модулей = 3

Число внутренних интерфейсных модулей = 6

Число модулей реализации = 7

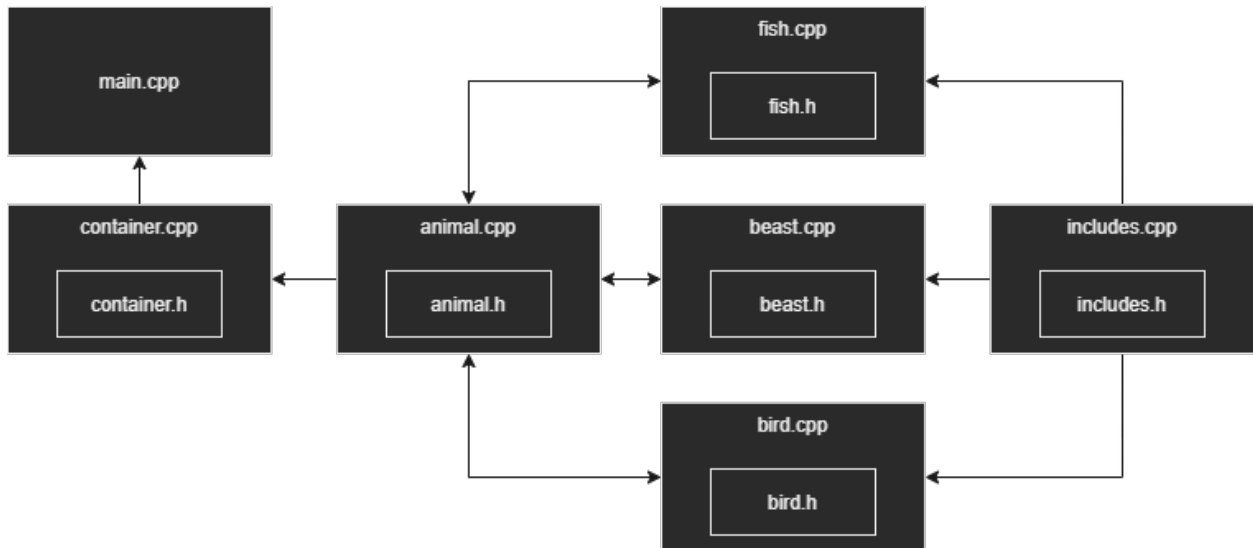
Общий размер исходных текстов программы
11 100 байт (на диске 24 576 байт)

Размер исполняемого файла
40 680 байт (на диске 40 960 байт)

	A	B	C	D
1	Тестовый файл	Количество элементов	Время C	Время C++
2	input01.txt	3	0.0001	0.0001
3	input02.txt	20	0.0004	0.0003
4	input03.txt	20	0.0003	0.0003
5	input04.txt	40	0.0004	0.0005
6	input05.txt	100	0.0009	0.0010
7	input06.txt	500	0.0041	0.0053
8	input07.txt	500	0.0043	0.0052
9	input08.txt	500	0.0044	0.0052
10	input09.txt	10 000	0.1087	0.1422
11	input10.txt	10 000	0.1135	0.1533
13	random	10	0.0002	0.0001
14	random	100	0.0005	0.0008
15	random	1 000	0.0061	0.0113
16	random	5 000	0.0382	0.0624
17	random	10 000	0.0876	0.1455

7. Схемы

Файловая схема



Классовая схема

Container
Length
Array [10'000]
Container()
~Container()
Out()
InRnd()
Value()
In()

Animal
Name
Weight
Random
StaticIn()
StaticInRnd()
Value()
virtual ~Animal()
virtual In()
virtual InRnd()
virtual Out()

Fish / Beast / Bird
Type
~Destructor()
In()
InRnd()
Out()

Схема памяти

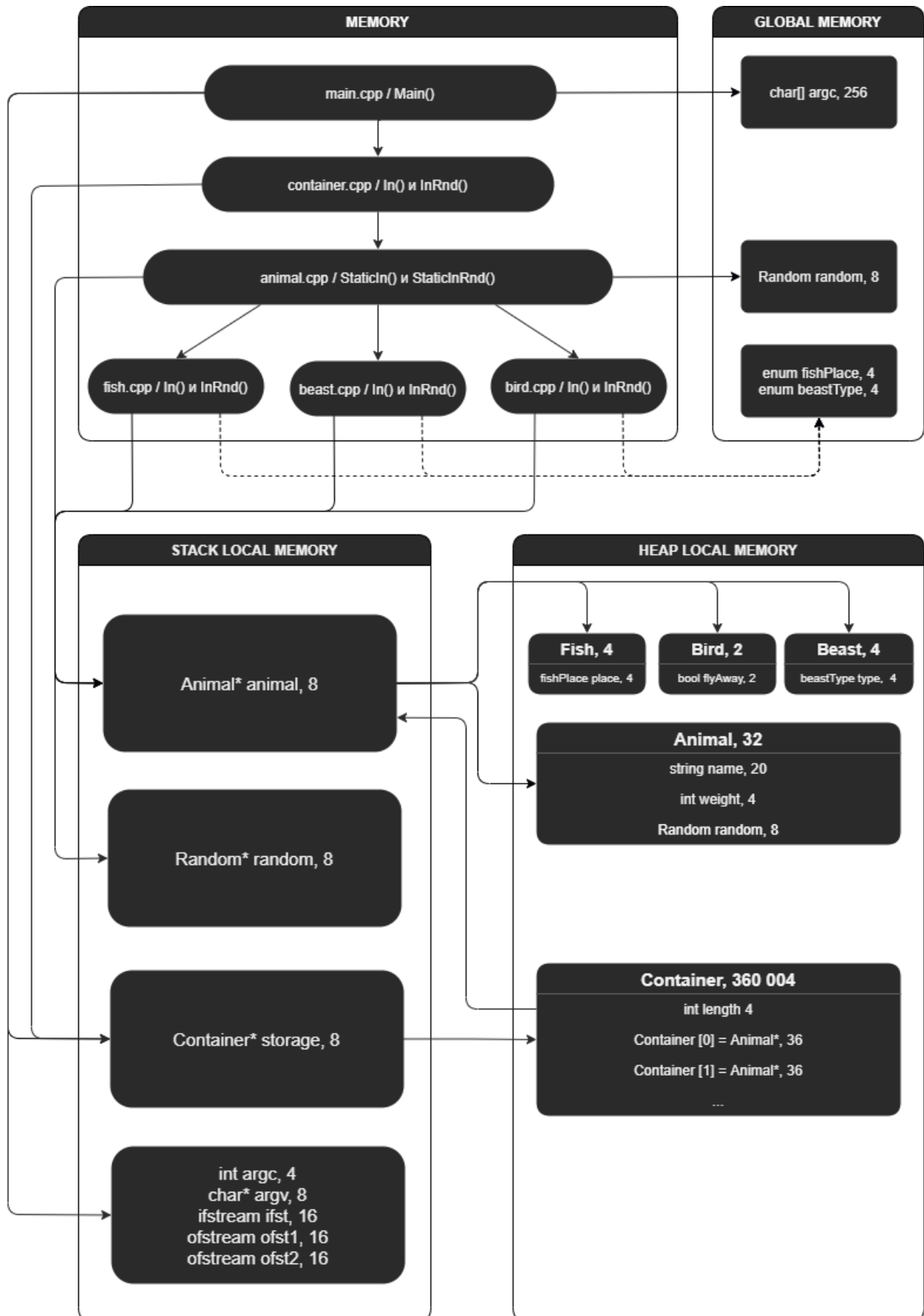


Таблица типов

Name		Size	
	int	4	
	double	8	
	char	1	
	void*	8	
	enum	4	
	Fish	1	
	bool flyAway	1	
	Bird	4	
	birdPlace place	4	
	Beast	4	
	beastType type	4	
	Animal	36	
	string name	20	char[] 16 int 4
	int weight	4	
	random random	8	int 4 int 4
	Beast / Fish / Bird	4	
	Container	360 004	
	int length	4	
	Animal* storage[10'000]	360 000	