

Описание полученного задания

Вариант по ABC

16 Милорадова Ксения Сергеевна 216

Variant number = 216

Number of task = 6

Number of function = 16

Обобщенный артефакт, используемый в задании:

6. Пассажирский транспорт

Базовые альтернативы (уникальные параметры, задающие отличительные признаки альтернатив):

1. Самолеты (дальность полета – целое, грузоподъемность – целое)
2. Поезда (количество вагонов – целое)
3. Корабли (водоизмещение – целое; вид судна – перечислимый тип = (лайнер, буксир, танкер)

Общие для всех альтернатив переменные:

1. Скорость – целое
2. Расстояние между пунктами отправления и назначения – действительное

Общие для всех альтернатив функции:

Идеальное время прохождения пути (действительное число)

После размещения данных в контейнер необходимо осуществить их обработку в соответствии с вариантом задания. Обработанные данные после этого заносятся в отдельный файл результатов. Необходимо реализовать одну из следующих функций:

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

Структурная схема изучаемой архитектуры ВС с размещенной на ней разработанной программы

Таблица типов

Название типа	Размер (бит)
int	4

double	8
clock_t	8
string	32
class Airplane distance: double speed: int flightRange: int loadCapacity: int	20 8[0] 4[8] 4[12] 4[16]
class Train distance: double speed: int numberOfWagons: int	16 8[0] 4[8] 4[12]
enum type	4[0]
class Ship distance: double speed: int displacement: int t: type	20 8[0] 4[8] 4[12] 4[16]
class Transport distance: double speed: int	12 8[0] 4[8]
class Container len: int Storage: *Transport[10000]	120004 4[0] 120000[4]
class Random first: int last: int	8 4[0] 4[4]
ofstream	512
ifstream	520

Глобальная память

Пусто

Память программы

main(int argc, char *argv[]) argc: int argv: char** start: clock_t c: Container outPath: string ifst: ifstream size: int ofst1: ofstream fin: ifstream end: clock_t	4[0] 8[4] 8[12] 120004[20] 32[120024] 520[120544] 4[120548] 512[120552] 520[121064] 8[121072]
errMessage1()	
Container:: In(ifstream &ifst)	

ifst: ifstream len: int storage: *Transport[10000]	520[0] 4[520] 120000[524]
Container:: InRnd(int size) len: int storage: *Transport[10000] size: int	4[0] 120000[4] 4[120004]

Требуемые метрики, определяющие характеристики программы, для различных тестовых прогонов

Число интерфейсных модулей: 6

Число модулей реализации: 5

Общий размер: 24KB

Полученный размер исполняемого кода: 170KB

Время выполнения программы для различных тестовых наборов данных:

Количество элементов	Общее время работы программы (секунды)
1	0.000871
10	0.000847
100	0.001416
1000	0.008329
10000	0.166872

Данные, демонстрирующие сравнение с характеристиками ранее разработанных программ

Количество элементов	Общее время работы программы (секунды)	
	Первая реализация	Вторая реализация
1	0.001179	0.000871
10	0.001458	0.000847
100	0.00126	0.001416
1000	0.007807	0.008329
10000	0.193598	0.166872
Число интерфейсных модулей	6	5
Число модулей реализации	5	5
Общий размер (KB)	24	23
Полученный размер исполняемого кода (KB)	170	322