Национальный исследовательский университет «Высшая школа экономики»

Домашнее задание №1

Архитектура вычислительных систем

Пояснительная записка

Исполнитель

студент группы БПИ205

Колб Илья Валерьевич

Задание

Вариант: 173

Номер задачи: 5

Номер функции: 13

Составление программы по условию:

* Обобщенный артефакт, используемый в задании: квадратная матрица с действительными числами.
* Базовые альтернативы (уникальные параметры, задающие отличительные признаки альтернатив):

1. Обычный двумерный массив.
2. Диагональная (на основе одномерного массива)
3. Нижняя треугольная (одномерный массив с формулой пересчета)

* Общие для всех альтернатив переменные:

1. Размерность (целое число)

* Общие для всех альтернатив функции

1. Вычисление среднего арифметического (действительное число)

* Расположение всех альтернатив в едином контейнере.
* Упорядочивание элементов контейнера по убыванию используя Shaker Sort. В качестве ключей для сортировки и других действий используются результаты функции, общей для всех альтернатив.
* Формат ввода:

1. Готовые тестовые файлы (./matrices\_dz1 -f <input file with data> <output file> <output sorted file>) input file:
2. Первый аргумент тип матрицы число от 1 до 3 (1 – обычная, 2 – диагональная, 3 – нижнетреугольная).
3. Второй аргумент общий для всех – размерность (натуральное число).
4. Третий аргумент зависящий от первых двух – список элементов в матрице (для обычной n\*n элементов, где n – размерность (действительные числа), для диагональной n элементов, где n – размерность (действительные числа), для нижнетрегольной n\*(n + 1)/2, где n – размерность (действительные числа))
5. Генерация случайных значений (./matrices\_dz1-n <number of figures> <output file> <output sorted file>)

Структурная схема изучаемой архитектуры ВС с размещенной на ней разработанной программы

Таблица типов

|  |  |
| --- | --- |
| int  double | 4 байта [0]  8 байт [4] |
| struct Common | 4 + 8\*dimension\* dimension байт |
| int dimension  double \*\*matrixArray | 4 байта [0]  8\*dimension\* dimension байт [4] |
| struct Diagonal | 4 + 8\*dimension байт |
| int dimension  double \*\*matrixArray | 4 байта [0]  8\*dimension байт [4] |
| struct Triangularn | 4 + 4\*dimension\*(dimension + 1) байт |
| int edge  double density | 4 байта [0]  4\*dimension\*(dimension + 1) байт [4] |
| Struct Shape | 20 + 12\*dimension\*(dimension + 1) ) байт |
| enum key  key k  Common s  Diagonal p  Triangularn t | 4 байта [0]  4 байта [4]  4 + 8\*dimension\* dimension байт [8]  4 + 8\*dimension байт [12 + 8\*dimension\* dimension]  4 + 4\*dimension\*(dimension + 1) байт [8\* (dimension\*(dimension + 1) + 2)] |
| Struct Container | 780008 байт |
| enum max\_len  int len  Shape cont | 4 байта [0]  4 байта [4]  52 \* 15000 байт (780000 байт) [8] |

Память программы

|  |  |
| --- | --- |
| main(int argc, char \*argv[])  int argc  char \*argv  Container c  int size | 4 байта [0]  8 байт [4]  780008 байт [12]  4 байта [780020] |
| void ErrMessage1 |  |
| void ErrMessage2 |  |
| void Init(Container &c) |  |
| void In(Container &c, FILE \*file) |  |
| void InRnd(Container &c, int size) |  |
| void ShakerSort(Container &c, int size) |  |
| bool swapped  int start, end, i | 1 байт[0]  3 \* 4 байта (12 байт)[1] |
| void Swap(Matrix &m1, Matrix &m2) |  |
| void Out(Container &c, FILE \*fileOutout) |  |
| Int i | 4 байта[0] |
| void Clear(Container &cont) |  |
| Int i | 4 байта[0] |

Стек вызовов (| - вызов одной из функций, ? – поведение программы при вызове функции)

|  |
| --- |
| main  ErrMessage1? – ~~main~~  Init  ~~Init~~  In (error – ~~main~~) | InRnd | ErrMessage2? – ~~main~~  ~~In~~ | ~~InRnd~~  Out  ~~Out~~  ShakeSort  ~~ShakeSort~~  Out  ~~Out~~  Clear  ~~Clear~~  ~~main~~ |

Характеристики программы

Интерфейсные модули: 6

Модули реализации: 7

Общий размер исходных тестов: 5,4 Мб

Общий размер результатов тестов: 1,3Мб

Результаты тестов (в секундах)

|  |  |
| --- | --- |
| Количество элементов | Ввод из файла |
| 5 | 0.000723 |
| 10 | 0.001385 |
| 100 | 0.008016 |
| 1000 | 0.245731 |
| 10000 | 21.507630 |
| 3 (некорректные) | 0.000593 |
| 1 (некорректный) | Вывод исключения |

Различия во времени не зависят от типа ввода, минимальные различия может выдавать сортировка, так как генерируемые данные могут быть частично упорядочены.

Заключение

Было разработано приложение по заданному ТЗ, а также проведен полный анализ структуры, памяти и времени работы.