

Задание 9 с обработкой данных в контейнере типа 13 Shaker (Cocktail sort)

В данном задании использована архитектура статически типизированного языка, ориентированного на процедурный подход. В основной структуре шифр хранятся признак альтернативы, расшифрованная строка, а также union структур шифров трёх типов. Отличительной особенностью данной архитектуры по отношению к заданиям 2 и 3 является факт, что расшифрованная а не зашифрованная строка принадлежит объекту обобщённого шифра, а так же тот факт что первичная генерация происходит для зашифрованной строки, а так же появление методов расшифровки по параметрам шифра в файле cipher. Это произошло как раз из-за реорганизации данных в соответствии с процедурным подходом. Так же можно отметить более долгое время разработки при не сильно изменившемся времени работы программы.

Файлы:

1. main.cpp – 1.9Kb – Основная логика программы. Заполняет контейнер и выполняет действия с ним.
2. cipher.cpp(4.1Kb)/ cipher.h(802B) – Структура Шифра. Хранит признаки альтернатив, а так же выполняет подсчёт общих данных по частным данным шифра.
3. cycleSwap.cpp(775B)/cycleSwap.h(387B) – Шифр сдвига на n
4. numberSwap.cpp(1.1Kb)/numberSwap.h(403B) – Шифр замены на числа
5. pointerSwap.cpp(1.4Kb)/pointerSwap.h(413B) – Шифр замены буквы на другую букву
6. container.cpp(2Kb)/container.h(976B) – Класс контейнера. Содержит все методы заполнения и очистки контейнера.

Решение также содержит тесты с примерами выводов. testA.txt и outA1.txt outA2.txt (A - число)– соответственно файлы входные и выходные. Некоторые тесты также проверяют некорректную работу программы. Файлы outx1.txt и outx2.txt – файлы вывода программы с максимальным количеством элементов в контейнере (10000 генерация и заполнение случайные).

Формат запуска:

Для случайной генерации:

```
./task01 -n [N – number of elements] [file out1] [file out2]
```

Для чтения из файла:

```
./task01 -n [file input] [file out1] [file out2]
```

Ввод можно посмотреть в тестах, для общего случая:

K – (определяет тип шифра 1 – cycleSwap 2 – numberSwap 3 - pointerSwap)

encryptedScript – (шифрованная строка)

Далее вводится признак шифра (n – число сдвига, или таблица замещения символов)

Stack
main
Init
In/InRnd
In/InRnd(by each cipher)

Out
CocktailSort
Out (by each cipher)

Heap
"./task01"
"-n"/"-f"
"input.txt"
"output1.txt"
"output2.txt"

Интерфейсных модулей – 5

Модулей реализации – 6

Время выполнения:

2 элемента – 6ms

20 элементов – 21ms

10000 элементов – 3m30s

Можно заметить, что время работы программы довольно большое. Это происходит из-за большого количества работы со строками, а также из-за решения получения строк лишь по зашифрованной и расшифрованной строке, реализация расшифровки экономит необходимое количество входных данных, но добавляет времени работы программы.