

СЕМИНАР 20 НОЯБРЯ

1. Напишите собственную функцию-связыватель `compose2`, получающую на вход бинарный функтор $F(x_1, x_2)$ и два унарных функтора $f_1(x)$ и $f_2(x)$ и возвращающую унарный функтор $F(f_1(x), f_2(x))$.
2. Заполните дек случайными целыми числами в интервале $[0, N)$. Скопируйте этот дек в выходной поток, а затем скопируйте в выходной поток те элементы дека, которые удовлетворяют условиям $a < x < b$. Используйте связыватели `bind1st`, `bind2nd`, `compose2` и адаптеры `less<int>` и `logical_and`.
3. Даны два текстовых файла. Найдите слова, встречающиеся в каждом из файлов. Найдите слова, встречающиеся только в одном из файлов. Напишите результаты на экране по алфавиту. Воспользуйтесь алгоритмами стандартной библиотеки `set_intersection` и `set_symmetric_difference`.
4. Напишите программу, получающую на входе слово и печатающую все его анаграммы (слова, полученные из исходного слова перестановкой букв). Воспользуйтесь стандартным алгоритмом `next_permutation`.
5. Напишите алгоритм `bd_sort`, который мог бы получать на вход двунаправленные итераторы, а не только итераторы произвольного доступа. Алгоритм должен работать так: если эти итераторы в самом деле являются итераторами произвольного доступа, то вызывается обычный `sort`. В противном случае последовательность копируется в вектор, который сортируется с помощью `sort`, а затем отсортированный вектор копируется обратно.