Homework Assignment 2

Due Date: March 4, 2022, 23:59

EXERCISE 1. Suppose you have to process n advance bookings of rooms for a hotel with k identical rooms. Bookings contain an arrival date and a departure date. You have to find out whether there are enough rooms in the hotel to satisfy the demands.

Design an algorithm that solves this problem in time $O(n \log n)$.

Hint. Sort the set of all arrivals and departures and process it in sorted order. Choose the most appropriate sorting algorithm for this problem.

total points: 10

EXERCISE 2. It is easy to check to check whether an algorithm produces a sorted output. It is less easy to check whether the output is also a permutation of the input. However, for integers there exists a fast and simple algorithm:

(i) Show that $[e_1, \dots, e_{n_i}]$ is a permutation of $[e'_1, \dots, e'_{n_i}]$ iff the polynomial

$$P(x) \;=\; \prod_{i=1}^{n}(x - e_i) - \prod_{i=1}^{n}(x - e'_i)$$

in the variable x is identically zero.

(ii) For any $\varepsilon > 0$ let p be a prime with $p > \max\{n/\varepsilon, e_1, \dots, e_{n_i}, e'_1, \dots, e'_{n_i}\}$. The idea is to evaluate the above polynomial $P(x)$ modulo p for a random value $x \in [0, p-1]$.

Show that if $[e_1, \dots, e_{n_i}]$ is not a permutation of $[e'_1, \dots, e'_{n_i}]$, then the result of the evaluation is zero with probability at most ε.

Hint. A non-zero polynomial of degree n has at most n zeroes.

total points: 11

EXERCISE 3. Show how to reverse the order of elements on a stack S

(i) using two additional stacks;

(ii) using one additional queue;

(iii) using one additional stack and some additional non-array variables.

total points: 9

EXERCISE 4. Implement bucket sort on sequences represented by unbounded arrays: The input is a sequence of real numbers x in the range $a \leq x < b$. For a sequence of length n split $[a, b)$ into n equi-distant buckets B_i $(0 \leq i \leq n-1)$, i.e. intervals $B_i = [a_i, a_{i+1})$ with $a_i = a + i \cdot \frac{b-a}{n}$. Use insertion sort to insert each element x of the input sequence into a list ℓ_i, provided $x \in B_i$ holds, then concatenate the lists $\ell_0, \dots, \ell_{n-1}$ to obtain the sorted output list.

Programming instructions. Use the class template ALIST and create a derived class template BLIST with a member function `bucketsort`.

Testing instructions. Test your program on a list containing 500 random integers in the range $[-1000, 1000)$. Use `append` to create the list.

The test-file `test.cpp` is provided.

total points: 20