

Университет ИТМО

Факультет программной инженерии и компьютерной техники

Лабораторная работа №3
по «Алгоритмам и структурам данных»

Выполнил:

Студент группы Р3200

Ле Туан Зунг

Преподаватели:

Косяков М.С.

Тараканов Д.С.

Санкт-Петербург

2019

1067 – Disk tree

Input:

- + N : total number of distinct directory paths
- + Each path is listed once and consists of a number of directory names separated by a back slash ("\").

Request:

- + need to build the formatted directory tree.

How to solve:

idea : build doubly linked list to store formatted directory tree

+ structure about each node:

- One node to the next name in current directory.
- One node to the subdirectory.

+ for each paths we start from the root of formatted tree then search for name.

+ we just search while the current string is smaller than our directory name, and if exist in our tree yet , we need to add them into the tree.

1521 – War game 2

Input:

+ The war games are divided into N phases; and N soldiers, successively numbered from 1 to N , are marching round a circle one after another, i.e. the first follows the second, the second follows the third, ..., the $(N-1)$ -th follows the N -th, and the N -th follows the first.

+ At some phase, the circle is left by a soldier, who is marching K positions before the one, who left the circle at the previous phase.

+ A soldier, whose number is K , leaves the circle at the first phase.

Request:

+ need to update the remaining position between all 2 soldiers after each phase.

How to solve:

+ we used interval tree in this task. Each node of tree will be the number between 2 soldiers (soldier I and soldier J)

+ for example :

- Set a is the soldier who left in the previous phase.
- Used binary search to search for the next soldier who will leave in the current phase by comparing the number of soldier between soldier (some soldier) and a with k .
- Need update interval tree every phase.

1494 – Monobilliards

Input:

- + N : number of billiard balls N .
- + In the next N lines there are the numbers of the balls in the order in which the inspector took them out from the pocket.

Request:

- + the word "Cheater" if Chichikov could not pocket all the N balls in the right order, otherwise output "Not a proof".

How to solve:

- + Because :” the inspector several times came up to the table and took out from the table's pocket the last of the pocketed balls.” So this work as LIFO way .

+ we set :

- $a[]$ is the list of pocketed balls in the table’s pocket (working like a stack).
- $a[i]$ is the last element of array $a[]$.
- k is the number of the ball that the inspector took.

+ for each k :

- If($k > a[i]$) then add the next number until we meet k into $a[]$, then $\text{pop}(k)$ from the stack.
- If($k = a[i]$) then $\text{pop}(k)$.
- If ($k < a[i]$) then chichikov cheated.