

Санкт-Петербургский политехнический университет Петра Великого
Институт компьютерных наук и технологий
**Высшая школа интеллектуальных систем и суперкомпьютерных
технологий**

КУРСОВОЙ ПРОЕКТ

«Treap»

по дисциплине «Алгоритмы и структуры данных»

Выполнил студент
гр. 3530901/00002

Сергеева Е. О.

Руководитель

Степанов Д. С.

«23» мая 2022 г.

Санкт-Петербург
2022

ЗАДАНИЕ НА ВЫПОЛНЕНИЕ КУРСОВОГО ПРОЕКТА

студенту группы 3530901/00002 Сергеевой Екатерине

1. Тема проекта: Реализация декартова дерева на языке Java

2. Срок сдачи законченного проекта: 23.05.2022

3. Исходные данные к проекту: IDE: IntelliJ IDEA Community Edition 2020.1, JVM: OpenJDK 64-Bit Server VM by JetBrains s.r.o, Version Java: 11

4. Содержание пояснительной записки (перечень подлежащих разработке вопросов): введение, основная часть (текст программы, описание программы, испытания программы), заключение, список использованных источников.

Дата получения задания: «01» апреля 2022 г.

Руководитель

(подпись)

Д. С. Степанов

(инициалы, фамилия)

Задание принял к исполнению

(подпись студента)

Е. О. Сергеева

(инициалы, фамилия)

(дата)

Исходные данные к работе

Вариант – «Реализация Treap»

Реализовать Декартово дерево, которое реализовывает интерфейс Set.

Текст программы

<https://github.com/sergeevaeo/kurs>

Описание программы

Декартово дерево реализовано в виде класса на Java. Класс дерева реализовывает интерфейс Set. Также добавлены вспомогательные методы Merge(), Split(), deep(), Print(), toString.

Класс Treap

size()

Возвращает размер дерева.

clear()

Удаляет все узлы дерева

isEmpty()

Проверка существует ли дерево

contains()

Возвращает true, если элемент есть в дереве, иначе false

containsAll()

Возвращает true, если все элементы из переданной коллекции содержатся в дереве, иначе false.

find()

Возвращает дерево, в корне которого будет переданный элемент.

Merge()

Возвращает дерево, которое получено из двух переданных деревьев

Split()

Возвращает два дерева, которые получены разделением исходного

add()

Добавление узла в дерево.

addAll()

Добавление всех элементов из коллекции.

findParent()

Нахождение родителя узла, возвращает дерево, в корне которого родитель узла.

remove()

Удаление элемента из дерева.

removeAll()

Удаление из дерева тех элементов, которые находятся в коллекции.

iterator()

Итератор для дерева.

hasNext()

Возвращает true, если следующий элемент существует, иначе false.

next ()

Возвращает следующий элемент.

remove()

Удаляет элемент, который был вызван последним next().

toArray()

Элементы дерева возвращаются в виде массива элементов.

toArray()

Элементы дерева возвращаются в виде массива элементов определенного типа.

retainAll()

Удаляет те элементы, которых нет в переданной коллекции.

Вспомогательные методы:

deep()

Возвращает длину самой длинной ветки.

toString()

Возвращает дерево в виде строки.

Print()

Рисует дерево на консоли.

Испытания программы

Для программы написаны тесты (TreapTests), проверяющие корректность выполнения всех функций. Для каждого метода написаны свои тесты.

Пример вывода дерева на консоль:

```
|          |-----(87; 19)
|          |          |          |-----(72; -68)
|          |          |-----(70; -4)
|----- (66; 24)
(53; 71)
|----- (47; -9)
|          |----- (44; -85)
|          |          |          |----- (12; -90)
|          |          |----- (6; -85)
|          |          |----- (1; -92)
```

Заключение

В результате выполнения курсового проекта я реализовала Декартово дерево, которое реализовывало интерфейс Set. Также были написаны тесты для каждого из методов.

Список использованных источников

- https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D0%BA%D0%B0%D1%80%D1%82%D0%BE%D0%B2%D0%BE_%D0%B4%D0%B5%D1%80%D0%B5%D0%B2%D0%BE
- <https://algorithmica.org/ru/treap>
- <https://e-maxx.ru/algo/treap>
- https://acm.khpnets.info/wiki/%D0%94%D0%B5%D0%BA%D0%B0%D1%80%D1%82%D0%BE%D0%B2%D0%BE_%D0%B4%D0%B5%D1%80%D0%B5%D0%B2%D0%BE
- <https://informatics.msk.ru/mod/book/view.php?id=423>