

Обучающая программа по теме: «Дерево отрезков»

Смирнова Елизавета Сергеевна

Ярославский государственный педагогический университет

им.К.Д.Ушинского

Студент

Быкова Ирина Альбертовна

Ярославский государственный педагогический университет

им.К.Д.Ушинского

Ассистент кафедры ТИМОИ

Аннотация

Была разработана обучающая программа, которая поможет обучающемуся в понимании работы дерева отрезков для быстрого поиска минимума или суммы на отрезке и реализации задач через разреженную таблицу.

Ключевые слова: обучающая программа, дерево отрезков, разреженная таблица, поиск минимума на отрезке, поиск суммы на отрезке

Training program for learning about Segment tree

Smirnova Elizaveta Sergeevna

Yaroslavl State Teacher Training University named after K.D.Ushinsky

Student

Bykova Irina Al'bertovna

Yaroslavl State Teacher Training University named after K.D.Ushinsky

Assistant of the Department of Theory and Methods of Teaching Informatics

Abstract

A training software program had been developed to help pupils and students in understanding the operation of the Segment tree for a quick search of the minimum or the sum in the interval and the implementation of tasks using a sparse table.

Keywords: training program, segment tree, sparse table, search minimum on an interval, find the sum on the interval

1. Введение

В интернете и в книгах вы все чаще можете столкнуться с теорией по дереву отрезков. Не совсем понятные формулы могут оттолкнуть вас от изучения такой интересной темы. Что мы и исправим.

2. Обоснование цели и задач

Актуальность данной темы заключается в том, что необходимо найти значение какой-то функции на отрезках линейного массива и иметь возможность быстро изменять значения группы подряд идущих элементов. Такую задачу решают, используя дерево отрезков.

Целями данного проекта являются:

- 1) Познакомить обучающихся с деревом отрезков
- 2) Пошагово объяснить способ построения и составления дерева отрезков
- 3) Пошагово объяснить способ отыскания суммы или минимума на заданном отрезке
- 4) Пошагово объяснить способ изменения элементов в дереве отрезков
- 5) Познакомить обучающихся с разреженной таблицей
- 6) Пошагово объяснить способ реализации задач по дереву отрезков через разреженную таблицу
- 7) Провести контроль усвоенного материала

3. Методы достижения целей

Для реализации поставленной цели мною были изучены пути построения дерева отрезков и разреженной таблицы, правила нахождения минимума и суммы на отрезке и модификация дерева отрезка.

Для наглядного объяснения материала была разработана обучающая программа с теоретическим описанием, пошаговым объяснением и контролем.

4. Теория

4.1. Общая теория по дереву отрезков и разреженной таблице

Дерево отрезков — структура данных, позволяющая быстро изменять значения в массиве и находить некоторые функции от элементов одномерного массива. Для удобства дополним длину массива до степени двойки. В добавленные элементы массива допишем бесконечности (за бесконечностью стоит понимать, например, число, больше которого в данных ничего не появится). Дерево отрезков это двоичное дерево, в каждой вершине которого написано значение заданной функции на некотором отрезке.

Каждому листу будет соответствовать элемент массива с номером, равным порядковому номеру листа в дереве. А каждой вершине, не являющейся листом, будет соответствовать отрезок из элементов массива соответствующих листам-потомкам этой вершины.

Корень будет лежать в первом элементе массива, а сыновья i -ой вершины будут лежать в элементах с номерами $2i$ и $2i + 1$ – левый и правый соответственно.

Разреженная таблица — двумерная структура данных $ST[i][k]$, для которой выполнено следующее: $ST[i][k] = \min/\max(A[i, k-1], A[i+2k-1, k-1])$, где i столбец, а k строка данной таблицы. При этом при заполнении разреженной

таблицы $ST[i][0]$ будет равно изначальному массиву, на котором мы хотим найти минимум/максимум на отрезке.

То есть в такой таблице хранятся минимумы/максимумы на всех отрезках, длины которых равны степеням двойки.

Алгоритм построения дерева отрезков (пошаговый):

1. Дополнение количества элементов до степени двойки, при этом в новые записываются значения, которые не смогут повлиять на построение. Например, плюс бесконечность для поиска минимума, минус бесконечность - для максимума, нуль - для суммы.

2. Записывает значения в массив, начиная с $2k$ до $2k+1-1$ индекса.

3. Теперь идет само заполнение, начиная с индекса $2k-1$ до 1. Значения получаются из выбора минимума (максимума/суммы) из элементов $2k-1*2$ и $2k-1*2+1$.

4.2. Запрос минимума

Назовём фундаментальным отрезком в массиве такой отрезок, что существует вершина в дереве, которой он соответствует. На вход поступают два числа левая граница и правая, и мы должны посчитать сумму чисел на этом отрезке. Будем подниматься снизу, добавляя к ответу на каждом уровне, если надо, фундаментальный отрезок.

Заведём два указателя – l и r , с помощью которых будем находить очередные фундаментальные отрезки разбиения. Изначально установим l и r указывающими на листы, соответствующие концам отрезка запроса. Заметим, что если l указывает на вершину, являющуюся правым сыном своего родителя, то эта вершина принадлежит разбиению на фундаментальные отрезки, в противном случае не принадлежит.

Аналогично с указателем r – если он указывает на вершину, являющуюся левым сыном своего родителя, то добавляем её в разбиение. После этого сдвигаем оба указателя на уровень выше и повторяем операцию. Продолжаем операции пока указатели не зайдут один за другой.

Находя очередной фундаментальный отрезок, мы сравниваем минимум на нём с текущим найденным минимумом и уменьшаем его в случае необходимости. Асимптотика работы алгоритма – $O(\log n)$, т. к. на каждом уровне мы выполняем константное число операций, а всего уровней – $\log n$.

4.3. Модификация

Заметим, что для каждого листа есть ровно $\log n$ фундаментальных отрезков, которые его содержат – все они соответствуют вершинам, лежащим на пути от нашего листа до корня. При обновлении элемента изменяется значение в полуинтервале, который ему соответствует, а также во всех полуинтервалах, которые его содержат. Несложно заметить, что с точки зрения дерева это все вершины, лежащие на пути от соответствующей вершины до корня. Значит, при изменении элемента достаточно просто

пробежаться от его листа до корня и обновить значение во всех вершинах на пути по формуле $T[i] = \min(T[2i], T[2i + 1])$.

5. Результат

В результате работы над данной темой была разработана программа.

Описание программы.

Когда мы запускаем программу, мы видим название проекта и в меню наверху список тех действий, которые в данный момент доступны.

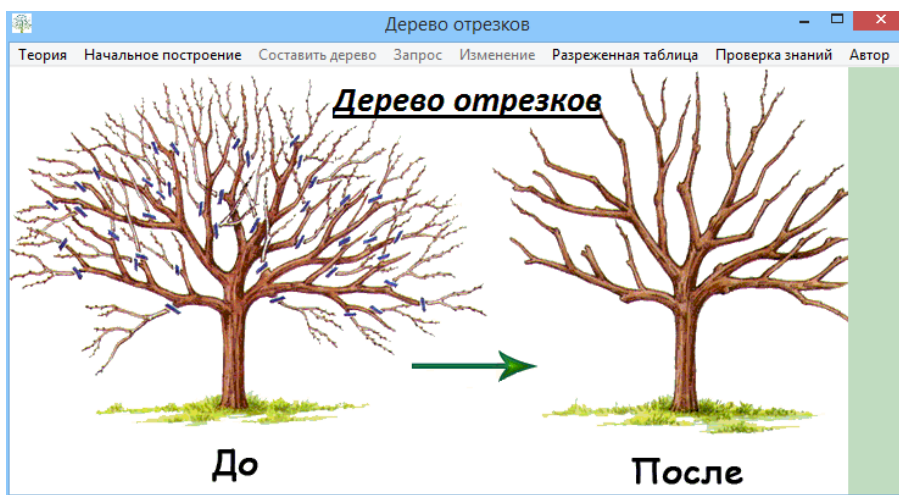


Рисунок 1 – Начальный запуск

При нажатии пункта меню теория, будет выбор посмотреть теорию общую для дерева отрезков, отдельно для составления суммы и минимума.

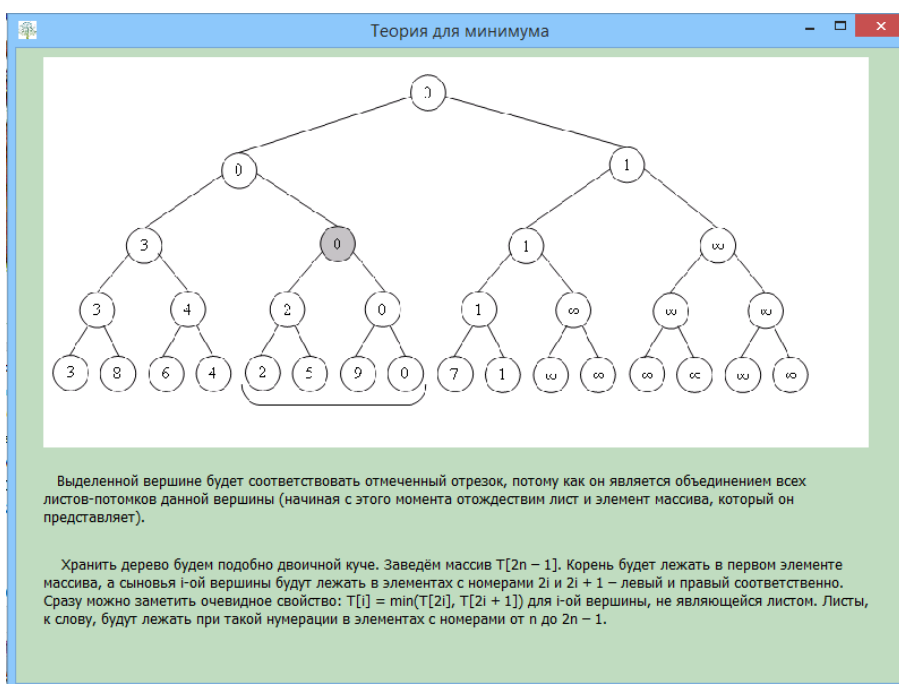


Рисунок 2 – Теория для минимума

Для пункта меню начальное построение возможно три вида создания дерева: считать из файла, ввод с клавиатуры и случайно. При этом заполняется только нижний ярус дерева.

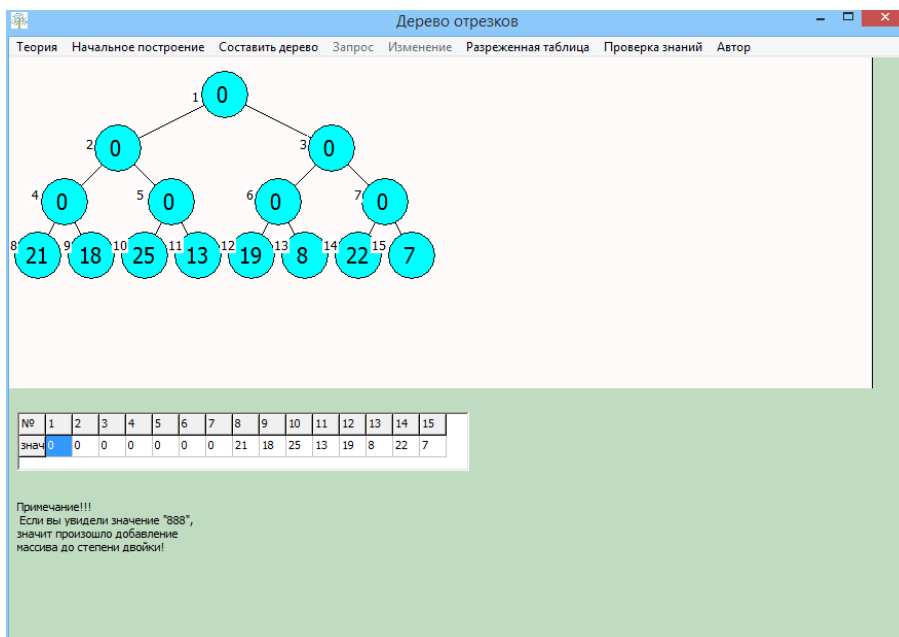


Рисунок 3 – Случайное построение дерева

После нажатия на любой из подпункта в пункте меню начальное построение, открывается для пользователя новый пункт меню составить дерево. При этом можно составить дерево минимума или суммы.

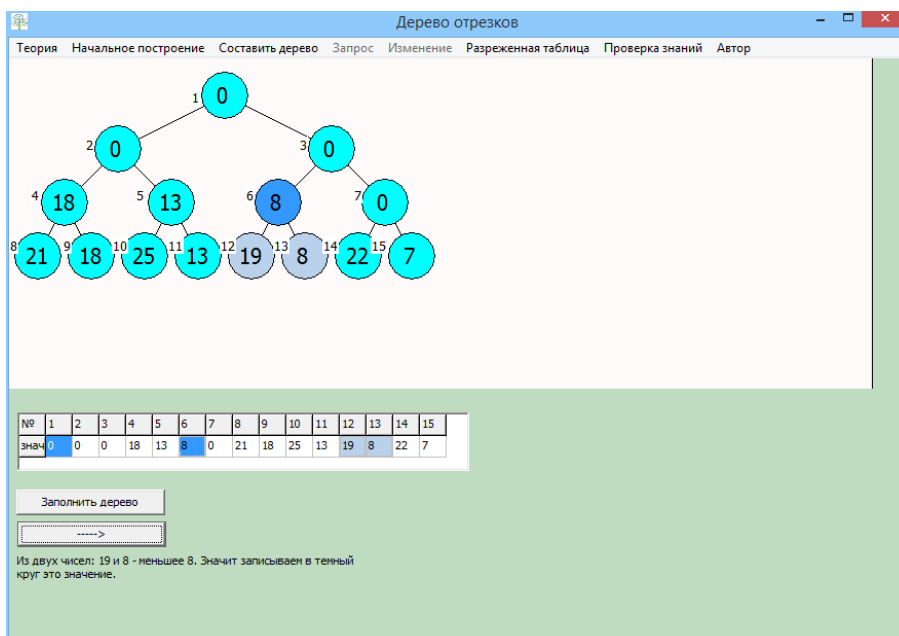


Рисунок 4 – Пошаговое заполнение дерева-минимума

При полном заполнении дерева откроется два новых пункта меню: запрос и изменение.

При нажатии пункта меню запрос, вам предложат выбрать на каком отрезке вы хотите найти минимум или сумму, в зависимости от того какой подпункт меню был выбран при составлении дерева.

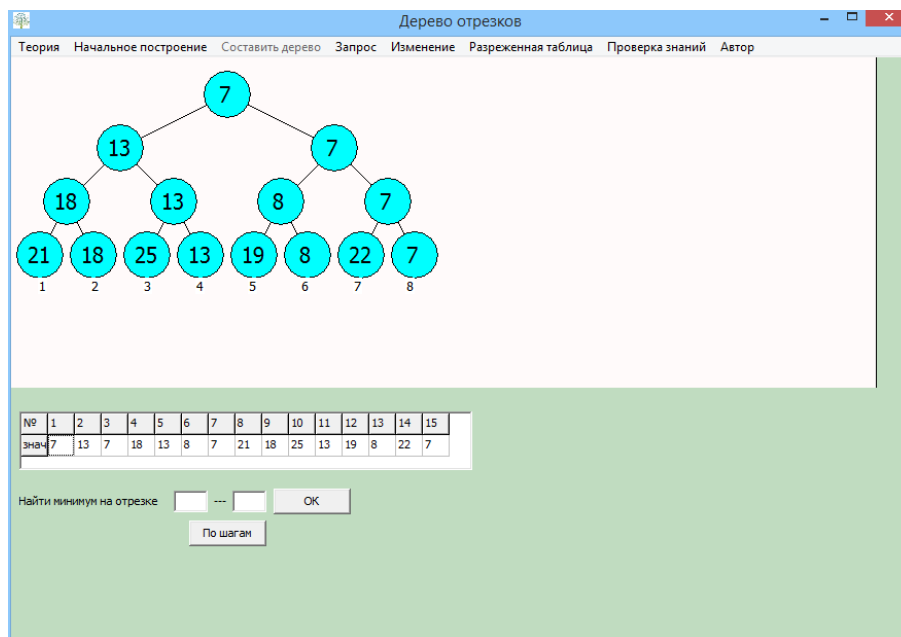


Рисунок 5 – Нажатие на пункт меню запрос

Вы можете пошагово просмотреть, как будет происходить выполнение запроса либо сразу вывести ответ.

При нажатии пункта меню изменение, вам предложат выбрать индекс элемента и новое значение для выбранного элемента.

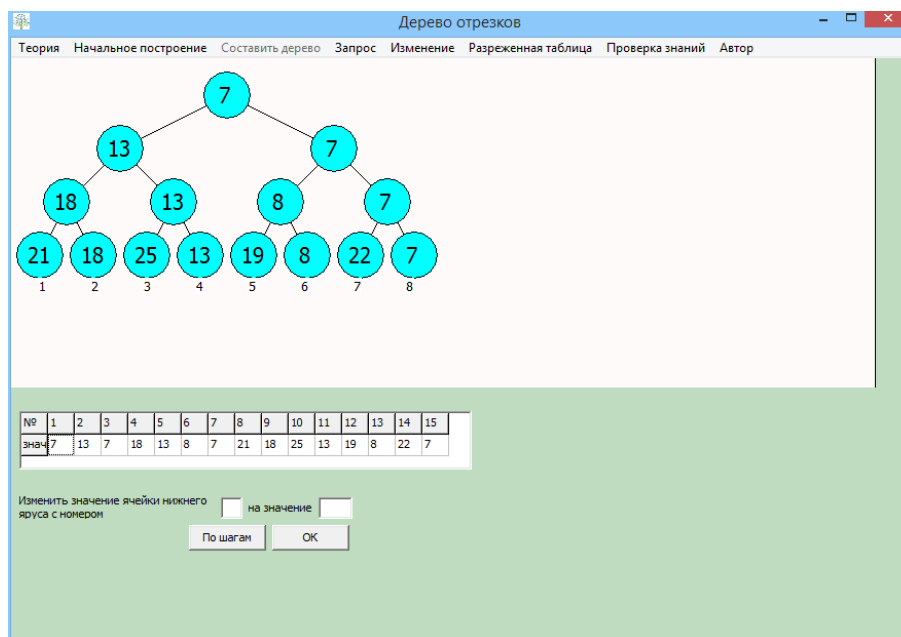


Рисунок 6 – Выбор пункта меню изменение

При нажатии пункта меню разреженная таблица, появится новое окошко с теорией, построением и проверкой знаний по разреженным таблицам.

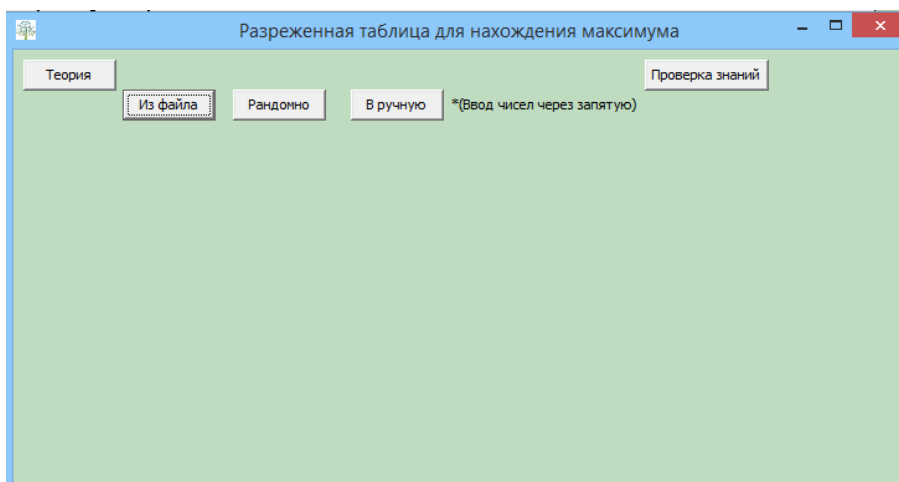


Рисунок 7 – Разреженная таблица

При нажатии кнопки теория, появится окошко с теорией по данной теме, при нажатии левой кнопки мыши, будет появляться новая информация, наглядно иллюстрирующая заполнение таблицы.

При нажатии одной из кнопок из файла, случайно или с клавиатуры произойдет первичное заполнение таблицы. Затем можно выполнить заполнение таблицы по шагам или сразу.

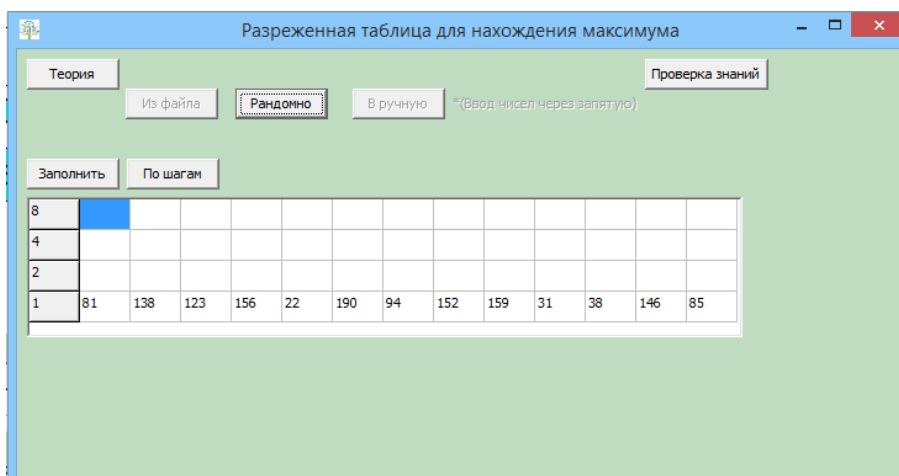


Рисунок 8 – Случайное заполнение разреженной таблицы

При нажатии проверка знаний появится новое окно, где пользователь сможет проверить свои приобретенные знания по разреженным таблицам. В тесте представлено 4 задания.

При нажатии проверка знаний в главном меню в окне дерево отрезков, появится окно с вводом данных пользователя, а именно Фамилия и Имя, и кнопка начать.

При нажатии пункта меню автор, появится окошко со сведениями об авторе программе.

Руководство пользователю

При начальном построении с клавиатуры нужно учесть, что программа рассчитана на количество вершин для нижнего яруса от двух до шестнадцати, значения не должно превышать 29. Ввод чисел производится через запятую.

При составлении запроса концы отрезка не должны быть пустыми, отрицательными, соответствовать значениям, не подписанным под нижним ярусом дерева.

При изменении элемента нижнего яруса дерева, нужно учесть, что индекс и новые значения индекса не должны быть пустыми, индекс не должен быть отрицательным.

При открытии теории в разреженных таблицах, нужно нажимать левую кнопку мыши для открытия новой информации.

6. Заключение

Демонстрационная программа Дерево отрезков решает одновременно несколько задач (обучение, контроль, теоретическое изложение материала), тем самым, помогая ученику полноценно изучить данную тему. Программа имеет простое меню и доступное объяснение решения. Тем самым, мы решили все поставленные задачи ранее.

Данная программа поможет ученикам усвоить материал и тут же проверить свои знания.

Библиографический список

1. Хабрахабр: [сайт]. URL: <https://habrahabr.ru/post/114980/>
2. Хабрахабр: [сайт]. URL: <https://habrahabr.ru/post/115026/>
3. Список лекций: [сайт]. URL: <https://brestprog.neocities.org/lections/segmenttree.html>
4. Университет ИТМО: [сайт]. URL: http://neerc.ifmo.ru/wiki/index.php?title=%D0%A0%D0%B5%D1%88%D0%B5%D0%BD%D0%B8%D0%B5_RMQ_%D1%81_%D0%BF%D0%BE%D0%BC%D0%BE%D1%89%D1%8C%D1%8E_%D1%80%D0%B0%D0%B7%D1%80%D0%B5%D0%B6%D0%B5%D0%BD%D0%BD%D0%BE%D0%B9_%D1%82%D0%B0%D0%B1%D0%BB%D0%B8%D1%86%D1%8B
5. Университет ИТМО: [сайт]. URL: http://neerc.ifmo.ru/wiki/index.php?title=%D0%94%D0%B5%D1%80%D0%B5%D0%B2%D0%BE_%D0%BE%D1%82%D1%80%D0%B5%D0%B7%D0%BA%D0%BE%D0%B2._%D0%9F%D0%BE%D1%81%D1%82%D1%80%D0%BE%D0%B5%D0%BD%D0%B8%D0%B5
6. MAXimal: [сайт]. URL: http://www.e-maxx-ru.1gb.ru/algo/segment_tree