Санкт-Петербургский Национальный Исследовательский Университет

Информационных Технологий, Механики и Оптики

Кафедра Систем Управления и Информатики

Лабораторная работа №1

Выполнили:

Лазуренко А.В., Маатук А.

Проверил

Мусаев А.А.

Санкт-Петербург,

2023

ЗАДАНИЕ 1

Файл с решением - binarysearch.py

Алгоритм бинарного поиска реализовали в функции bs(arr, x), которая принимает на вход список чисел (формат list()), среди которых нужно найти число, и само число (int()). Список arr представляет из себя набор заранее заданных чисел, идущих в порядке возрастания. Числа не заданы правилами какой-либо последовательности. На выходе функция возвращает индекс искомого числа в списке и количество итераций, за которое оно было найдено.

В переменной l рассматривается индекс левой границы рассматриваемого диапазона, в переменной h – правой, в переменной m – середины рассматриваемого диапазона. В переменной i храним количество итераций цикла. Согласно алгоритму бинарного поиска, в цикле проверяем: если искомое число больше, чем число из середины рассматриваемого диапазона, то диапазон смещается в правую часть списка, иначе в правую.

ЗАДАНИЕ 2

Для решения используется файл opros.csv с вопросами и информацией об ответах на них от о всех участников. Данные из файла извлекаются с помошью модуля csv в функциях parse\_answers(), которая возвращает список объектов класса Anket(answers, name) (list[Anket]), и get\_questions(), которая возвращает список объектов класса str() (list[str]).

В классе Anket инициализируются атрибуты answers типа list[str] и name типа str. С помощью объектов данного класса в дальнейшем происходит сравнение введенных пользователем данных с данными об ответах опрашиваемых.

В функции parse\_answers() происходит чтение файла, в объекте reader за ненадобностью удаляются столбцы «time» и «Имя Фамилия». В результате итерации цикла создается объект класса Anket и заносится в возвращаемый список.

В функции get\_questions() происходит повторное чтение файла, удаление ненужных столбцов и, в результате единственной итерации цикла, извлекаются вопросы для анкеты.

Функция check\_anket(checkrange, answers\_form, anket\_form) получает на вход диапазон сравниваемых элементов списка, список ответов, введенных пользователем и сравниваемую с ними анкету. Функция возвращает истину, если сравниваемые ответы совпадают и ложь в противном случае.

Процедура start\_dialog(questions, ankets) получает на вход список вопросов в формате list[str] и список анкет, извлеченных из файла opros.csv list[Anket]. На выходе процедура выводит результат опроса пользователя. В начале создается пустая форма, в которую заносятся ответы, введенные пользователем, и логическая переменная для контроля финального вывода. В цикле динамически происходит анкетирование пользователя и сравнение его ответов со всеми анкетами.

Так как вероятность полного совпадения анкет разных двух разных людей уменьшается с каждым следующим вопросом, то для точного определения личности пользователя по базе данных не обязательно задавать все вопросы. После каждого ввода ответа происходит сравнение уже введенных ответов с соответствующим количеством вопросов в анкетах каждого опрашиваемого из базы данных. Если существует только одна полностью идентичная анкета, то однозначно определяется личность пользователя и программа завершается. Если ни одна анкета не соответствует введенным ответам, то программа выводит соответствущее сообщение.

ЗАДАНИЕ 3

Граф построен в виде блок-схемы, представляющей из себя бинарное дерево. Структура графа основана на алгоритме определения личности пользователя по уникальному набору ответов. Граф построен с помощью бесплатного сервиса Programforyou. Так как экспортированное изображение имеет малое разрешение, предусмотрен вариант загрузки файла tree.json в редактор блок-схем для более комфортного просмотра графа. Инструкция загрузки описана в файле README.txt

Использемые источники

1. Programforyou. Онлайн редактор блок-схем. [Электронный ресурс] – https://programforyou.ru/block-diagram-redactor;

(Дата обращения 14.09.2023)