

Лабораторная работа 3

Линейные структуры данных

(20 баллов)

Выполните самостоятельно следующие задания и оформите отчет.

Требования по отчету:

Наличие титульного листа. Размер страницы должен соответствовать формату А4 (210x297), размеры полей: левое – 30 мм, правое – 10 мм, верхнее – 15 мм, нижнее – 20 мм. Шрифт Times new Roman, размер 14 pt полуторный междустрочный интервал. Выравнивание текста – по ширине, красная строка – 1,25 см, отступ слева и справа – 0 мм.

СВЯЗНЫЙ СПИСОК

№ 1

(3 балла)

Возьмите реализацию класса `UnorderedList` из лекционных материалов и выполните следующие доработки:

1. Реализуйте оставшиеся методы этого класса `append(item)`, `index(item)`, `insert(pos,item)`, `pop()`, `pop(pos)`. Требования к работе этих методов посмотрите в лекционной презентации.
2. Реализуйте метод `__str__`, чтобы содержимое списка отображалось в Python-стиле (в квадратных скобках).
3. Реализуйте метод “среза”. Он должен принимать два параметра (`start` и `stop`) и возвращать копию списка, начиная с позиции `start` и заканчивая (но не включая!) позицией `stop`.

Все выполненные доработки должны быть протестированы.

№ 2

(2 балла)

Напишите функцию `move_to_front` для чтения строк с клавиатуры и сохранения их в связном списке без повторов. Пользователь последовательно вводит с клавиатуры `n` строк. При вводе строки, которая раньше не встречалась, вставьте ее в начало списка. При вводе уже внесенной строки удалите ее из списка и снова вставьте в начало.

Такая функция реализует стратегию «сдвиг в начало», которая применяется в кешировании, при сжатии данных и т.д.

№ 3

(2 балла)

Напишите функцию, которая принимает в качестве аргумента первый узел односвязного списка, обращает порядок элементов списка (с разрушением исходного) и возвращает в

качестве результата первый узел полученного списка. Разработайте два варианта решения этой задачи: 1) итеративный и 2) рекурсивный.

№ 4

(3 балла)

Используя реализацию связного списка из задания № 1 (класс `UnorderedList`), реализуйте следующие линейные структуры (необходимый набор методов посмотрите в лекционной презентации):

1. Стек.
2. Очередь.
3. Дек.

Проведите тестирование реализованных линейных структур.

Продумайте и проведите показательный эксперимент, сравнивающий производительность стека, очереди или дека, основанных на списках Python, с реализациями на основе связанного списка. Например, для сравнения можно взять какую-либо операцию выбранной линейной структуры.

№ 5

(2 балла)

Реализуйте класс `DoubleList` для построения двухсвязных списков, в которых каждый узел содержит ссылку на узел, предшествующий ему в списке, и ссылку на следующий узел.

Класс должен иметь следующие методы: проверка списка на пустоту, поиск элемента, размер списка, вставка в начало, вставка в конец, удаление из начала, удаление из конца, вставка перед указанным узлом, вставка после указанного узла, удаление указанного узла.

Проведите тестирование методов реализованного класса.

Стек

№ 6

(2 балла)

В HTML у каждого тега существуют открывающая и закрывающая формы, которые должны быть сбалансированы, чтобы адекватно описывать веб-документ. Вот пример очень простого веб-документа:

```
<html>
  <head>
    <title>
      Example
    </title>
  </head>

  <body>
    <h1>Hello, world</h1>
  </body>
</html>
```

Обязательно укажите сделанные вами предположения и предоставьте любые вероятностные данные, которые нужно будет рассмотреть в качестве части сценария.