

1. Автобус

Автобус следует от остановки до остановки. На остановке пассажиры заходят через заднюю дверь, выходят — через переднюю. Реализовать структуру данных, хранящих информацию о пассажире:

- на какой остановке пассажир зашел в автобус;
- на какой остановке пассажир выходит из автобуса.

Алгоритм должен выполнять операции посадки в автобус, перемещения пассажира к передней двери перед выходом, выход пассажира из автобуса.

2. Телефонная книга

В телефонную книгу человек записывает контакты. На страницу телефонной книги записываются новые контакты, пока на ней есть место. Когда место заканчивается, запись ведется на следующей странице.

Реализовать две структуры данных, представляющую собой страницу из телефонной книги и саму телефонную книгу.

Алгоритм должен выполнять операции добавления контактов в книгу, вывод текущей страницы книги, перелистывание страниц.

3. Список контактов

В списке контактов телефона хранятся контакты владельца. Контакты отсортированы в лексикографическом порядке.

Реализовать структуру данных, представляющую список контактов. Контакты должны быть отсортированы в лексикографическом порядке.

Алгоритм должен выполнять операции добавления и удаления контактов, поиск контакта по первой букве имени.

4. Список покупок

Семья из трех человек формирует список покупок перед походом в магазин. Один из членов семьи идет в магазин и покупает из списка то, что есть в наличии в магазине.

Реализовать структуру данных, представляющую список покупок.

Алгоритм должен выполнять операции добавления наименования и количества товаров в список, если товар уже есть в списке, то добавляется

необходимое количество в уже существующую запись. Удаление из списка тех товаров, которые есть в наличии в магазине и были куплены.

5. Очередь в магазине

В магазине есть три кассы, на которых могут одновременно работать от 1 до 3 кассиров. Люди приходят и становятся в очередь. При появлении нового кассира, очередь должна перераспределиться.

Реализовать структуру данных, представляющую очередь на кассы.

Алгоритм должен выполнять операции по добавлению покупателей в очередь (добавление осуществляется естественным образом – где меньше людей, туда и становимся), по уходу расплатившихся покупателей (считать, что человек у кассы обслуживается за одну итерацию работы алгоритма), по перераспределению очереди, в случае появления еще одного/двух кассиров.

6. Книжный шкаф

У книголюба есть шкаф, где он хранит свои книги. Он постоянно пополняет свою коллекцию выставляя новые книги на свободные полки. Периодически он расставляет книги по алфавиту, чтобы поддерживать порядок в шкафу.

Реализовать две структуры данных, представляющие книжную полку и книжный шкаф.

Алгоритм должен выполнять добавление книги на место любой свободной полки, извлечение книги с полки, операцию «Уборка», выполняющую сортировку книг по алфавиту и выставление их так, чтобы они заполняли полки полностью, начиная с верхней.

7. Ханойская башня

Реализовать игру ханойская башня.

Структура данных должна представлять ханойскую башню.

Алгоритмы должны выполнять перестановку колец башни между тремя стержнями, проверять правильность хода, сигнализировать об успешном окончании. Реализовать исключение игрока, в случае неоднократного нарушения правил.

8. Посуда в ресторане

Реализовать процесс мытья посуды в ресторане. Мойщику посуды официанты приносят грязную посуду и кладут их в одну стопку. Мытую посуду мойщик складывает в другую стопку, и которой ее забирают повара.

Реализовать две структуры данных, представляющую мойщика посуды и стопки с посудой.

Алгоритмы должны выполнять добавление и удаление посуды из стопок. Если чистой посуды остается очень мало, необходимо позвать еще мойщика, если грязная посуда заканчивается – все мойщики идут отдыхать.

9. Play-list

Пользователи смартфонов формируют музыкальный play-list.

Реализовать структуру данных, представляющую play-list.

Алгоритмы должны выполнять добавление и удаление композиций в список, воспроизведение выбранной композиции, в соответствии с двумя режимами: случайное воспроизведение и последовательное воспроизведение.

10. KFC. So good

При заказе в KFC формируется электронная очередь. Если составляющие заказа готовы, то заказ собирается и отдается покупателю. Если нет, то пропускается и собирается следующий.

Реализовать структуру данных, представляющую электронную очередь.

Алгоритмы должны выполнять добавление в очередь и удаление из очереди при получении заказа, выбор заказа, который собирается на данный момент, исходя из порядка в очереди и готовности позиций заказа. Учесть что кухня периодически пополняет запас готовых позиций.