Зачет. Алгоритмы и структуры данных

1. Понятие структуры данных. Виды структур данных, их классификация

Структура данных — программная единица, позволяющая хранить и обрабатывать множество однотипных и/или логически связанных данных в вычислительной технике.

Рекурсивные структуры данных

* строчные
  + дек
  + стек
  + очередь
* списковые

Список - набор элементов, расположенных в определенном порядке.

Список очередности - список, в котором последний поступающий эле-

мент добавляется к нижней части списка.

Список с использованием указателей - список, в котором каждый элемент содержит указатель на следующий элемент списка.

2. Строчные структуры данных. Очередь. Стек. Дек

Очередь - это такой тип данных, при котором новые данные располагаются следом за существующими в порядке поступления; поступившие первыми данные при этом обрабатываются первыми(это последовательность, в которую включают элементы с одной стороны, а исключают — с другой)

Стек - это линейный список, в котором все включения и исключения делаются в одном конце списка.

Дек - это линейная структура (последовательность), в которой операции включения и исключения элементов могут выполняться как с одного, так и с другого конца последовательности.

3. Понятие и свойства алгоритма

[Алгоритм](http://ru.wikipedia.org/wiki/%D0%90%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC) — набор инструкций, четко описывающих порядок действий исполнителя для достижения результата решения задачи за конечное число действий.

[Свойства алгоритмов](http://gkl-it.blogspot.ru/2012/11/blog-post_23.html):

1. [Дискретность](http://ru.wikipedia.org/wiki/%D0%94%D0%B8%D1%81%D0%BA%D1%80%D0%B5%D1%82%D0%BD%D0%BE%D1%81%D1%82%D1%8C) (алгоритм должен представлять решение задачи как последовательность некоторых простых шагов);
2. [Детерминированность](http://ru.wikipedia.org/wiki/%D0%94%D0%B5%D1%82%D0%B5%D1%80%D0%BC%D0%B8%D0%BD%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%BE%D1%81%D1%82%D1%8C) (алгоритм должен в каждый конкретный момент времени однозначно определять следующий шаг исходя из состояния системы);
3. Понятность (алгоритм должен включать только те команды, которые исполнитель может выполнить);
4. Конечность (алгоритм при корректных входных данных должен заканчиваться и выдавать результат);
5. Универсальность (алгоритм должен быть применим к разным наборам входных данных);
6. Результативность (алгоритм должен завершаться с каким-то результатом);

4. Построение алгоритма. Теорема Бёма-Якопини

5. Формы представления алгоритмов

6. Классификация и критерии эффективность алгоритмов поиска и сортировки

7. Сортировка вставкой

8. Сортировка простым обменом и шейкерная сортировка

9. Сортировка выбором

10. Метод Шелла. Сортировка "расческой"

11. Алгоритм Хоара

12. Сортировка слиянием

13. Турнирная и пирамидальная сортировки

14. Сортировка подсчетом, блочная и поразрядная сортировки

15. Последовательный поиск

16. Бинарный поиск

17. Фибоначчиев поиск. Метод золотого сечения

18. Интерполяционный поиск

19. Поиск по бору

20. Поиск хешированием

21. Сложность и эффективность алгоритмов. Временная и емкостная сложность

22. Понятие, типы и классификация функций сложности

23. Правила преобразования асимптотических функций сложности

24. Функции сложности основных управляющих конструкций

25. Теоретическая оценка функции сложности алгоритма

26. Экспериментальная оценка функции сложности алгоритма

27. Понятие итерации и рекурсии. Итеративные и рекурсивные алгоритмы

28. Виды рекурсии

29. Хвостовая рекурсия

30. Глубина рекурсии. Вызов функций на стеке

31. Преобразование итерации к рекурсии

32. Преобразование рекурсии к итерации

33. Применение рекурсии. Метод "разделяй и властвуй"

34. Применение рекурсии. Поиск с возвратом

35. Понятие графа. Формы представления графов в ЭВМ

36. Поиск минимального остовного дерева алгоритмом Краскала

37. Поиск минимального остовного дерева алгоритмом Прима

38. Поиск кратчайших путей на графах. Алгоритм Дейскстры

39. Поиск кратчайших путей на графах. Алгоритм Роя-Флойда-Уоршелла

40. Поиск кратчайших путей на графах методом динамического программирования

41. Бинарное дерево поиска (БДП). Обход бинарного дерева

42. Нахождение минимального, максимального, предыдущего и следующего элементов в БДП

43. Вставка и удаление элемента из БДП

44. B-деревья. Принципы построения, вставка и удаление элементов

45. Модификации B-деревьев: B+-дерево, B\*-дерево, 2-3-дерево, (a,b)-дерево

46. Красно-черные деревья. Принципы построения и балансировки

47. АВЛ-деревья. Принципы построения и балансировки

48. Понятие и принцип построения кучи. Двоичная куча

49. Операции с кучами: построение кучи, восстановление свойства кучи

50. Операции с кучами: извлечение максимума (минимума), добавление элемента

51. Эвристические алгоритмы. Задача поиска пути в лабиринте

52. Метод ветвей и границ. Задача коммивояжера

53. Клеточные автоматы

54. Машина Тьюринга

+ Задачи по темам семинаров!