**Лабораторная работа №1**

Полустатические структуры данных. Стеки.

Цель работы: Приобрести навыки работы с последовательными списками типа стек.

Подготовка к лабораторной работе

При подготовке к лабораторной работе следует повторить следующие вопросы: понятие стека, принцип организации стека, основные операции над стеком, размещение стека в памяти ЭВМ, логическая и физическая структуры стека. В соответствии с заданием на лабораторную работу разработать схему алгоритма и составить программу на языке СИ.

Порядок выполнения работы

1. Разработать программу по заданию, выбранному в соответствии с вашим вариантом.

2. Создать исходный модуль с текстом программы.

3. Произвести отладку программы.

4. Выполнить программу, провести анализ результатов и, убедившись в правильности решения.

Содержание отчета

Отчет по лабораторной работе должен содержать:

1. Задание к лабораторной работе.

2. Схема алгоритма решения задачи.

3. Программу на языке СИ.

4. Результаты выполнения программы.

Контрольные вопросы

1. Понятие стека.

2. Размещение стека в памяти ЭВМ.

3. Основные операции над стеком.

4. Дескриптор стека.

Задание к лабораторной работе

1. Записать в стек 10 целых чисел. Переписать во второй стек четные числа, а в третий – нечетные.

2. Записать в стек предложение. Переписать во второй стек предложение таким  образом, чтобы слова были записаны в нем в обратном порядке.

3. Записать в стек числа, как положительные, так и отрицательные. Удалить из стека отрицательные числа, записав их в другой стек. Вывести на печать содержимое исходного и результирующего стеков.

4. Дан стек, содержащий целые числа. Выдать на печать четные значения стека. Нечетные значения записать в массив.

5. Записать в стек последовательность целых чисел, оканчивающуюся нулем. Переписать во второй стек числа меньше 10.

6. Используя стек, провести проверку строки на симметричность относительно символа “\*”.

7. Даны два стека, содержащие по 4 вещественных числа. Записать их в третий стек, чередуя числа из первого и второго стеков, причем числа первого стека разместить в обратном порядке.

8. Создать два стека, записав в них целые числа. Записать в третий стек числа, равные попарно просуммированным значениям из первого и второго стеков. Вывести на печать содержимое всех трех стеков.

9. Записать в стек произвольную последовательность символов. Переписать в другой стек из первого гласные буквы русского алфавита. Содержимое второго стека вывести на печать.

10. Записать в стек произвольное число элементов. Вывести на печать третий элемент стека. Содержимое стека должно остаться без изменения.

11. Создать стек произвольной длины. Удалить из стека третий с конца элемент, присвоив его значение параметру Х. Вывести на печать содержимое исходного стека, а также значение параметра Х.

12. Дан стек, содержащий предложение, оканчивающееся точкой. Переписать во второй стек последнее слово предложения. Вывести на печать содержимое двух стеков.

13. Даны два стека, содержащие по пять целых чисел. Записать их содержимое в третий стек, чередуя числа первого и второго стеков.

14. Записать в стек последовательность символов. Выдать на печать последовательность с сохранением исходного порядка.

15. Дан стек, содержащий последовательность символов. Переписать символы в обратном порядке во второй стек.

16. Создать два стека, записав в них целые числа. Записать в третий стек числа, равные разностям соответствующих значений из первого и второго стеков. Вывести на печать содержимое всех трех стеков.

17. Записать в стек произвольную последовательность символов. Переписать в другой стек из первого небуквенные символы. Содержимое второго стека вывести на печать.

18. Записать в стек произвольное число элементов. Вывести на печать каждый третий элемент стека. Содержимое стека должно остаться без изменения.

19. Записать в стек последовательность целых чисел, оканчивающуюся нулем. Переписать во второй стек числа меньше среднего арифметического.

20. Даны два стека, содержащие по 6 вещественных чисел. Записать их в третий стек, чередуя числа из первого и второго стеков.

**Лабораторная работа №2**

Полустатические структуры данных. Очереди.

Цель работы: Приобрести навыки работы с очередями.

Подготовка к лабораторной работе

При подготовке к лабораторной работе следует повторить следующие вопросы: понятие очереди, принцип организации очередей, основные операции над очередями, размещение очереди в памяти ЭВМ, логическая и физическая структуры очереди. В соответствии с заданием на лабораторную работу разработать схему алгоритма и составить программу на языке СИ.

Порядок выполнения работы

1.Разработать программу по заданию, выбранному в соответствии с вашим вариантом.

2.Ввести текст программы.

3.Произвести отладку программы.

4.Выполнить программу, провести анализ результатов и, убедившись в правильности решения, предъявить их преподавателю.

Содержание отчета

Отчет по лабораторной работе должен содержать:

1.Задание к лабораторной работе.

2.Схема алгоритма решения задачи.

3.Программу на языке СИ.

4.Результаты расчета (листинг).

Контрольные вопросы

1.Понятие очереди.

2.Физическая и логическая структуры очереди.

3.Основные операции над элементами очереди.

4.Понятия слота и дескриптора.

Задание к лабораторной работе

1.Записать в очередь произвольную последовательность элементов. Вывести на печать нечетные компоненты очереди.

2.Записать последовательность цифр в очередь в порядке возрастания. Ввести еще одно число и поместить его в очередь, не нарушая принципа упорядоченности элементов.

3.Вводить последовательность символов, помещая её в стек. Если в строке встречается гласная буква русского алфавита, то содержимое стека переписывается в очередь. Содержимое очереди периодически выводить на экран. По окончании ввода распечатать содержимое стека.

4.Из данной очереди сформировать очередь, содержащую только буквы латинского алфавита в исходном порядке. Вывести на печать содержимое двух очередей.

5.Записать в очередь средние баллы групп (кол-во групп вводится с клавиатуры). Вывести на экран содержимое очереди в исходном порядке. Определить группу с максимальным средним баллом.

6.Сформировать пустую очередь и заполнить её сведениями о группах факультета (номер группы, кол-во студентов). Удалить из очереди группу с наименьшим количеством студентов.

7.Сформировать непустую очередь из последовательности целых чисел. Исключить элементы, которые нарушают условие возрастания.

8.Из стека, содержащего последовательность символов, переписать в очередь символы русского алфавита.

9.Ввести символьную последовательность, оканчивающуюся точкой. Гласные буквы латинского алфавита включить в очередь. Если в очередь включено более 5 элементов, лишние исключить.

10.Из очереди удалить каждый третий элемент, записав их в другую очередь.

11.Дан стек. Создать очередь из четных элементов стека. Вывести на печать содержимое стека и очереди.

12.Дана очередь из элементов, упорядоченных по убыванию. Добавить в очередь элемент, не нарушая упорядоченности. Вывести на печать исходную и результирующую очереди.

13.Переписать содержимое стека в очередь.

14.Даны две очереди, содержащие символы. Записать их содержимое в третью очередь, чередуя символы второй и первой очередей.

15.Даны две очереди. Вывести на печать элементы той очереди, которая длиннее.

16.Строковая очередь считывается из файла и выводится на печать.

17.Записать в очередь произвольную числовую последовательность элементов. Переписать в другую очередь компоненты кратные трем.

18.Ввести символьную последовательность, оканчивающуюся точкой. Гласные буквы латинского алфавита включить в первую очередь, а остальные символы – во вторую. Вывести на печать элементы созданных очередей.

19.Создать очередь из целых чисел. Удалить из этой очереди числа кратные пяти. Вывести полученную очередь на печать.

20.Создать очередь из целых чисел. Удалить из этой очереди числа больше среднего значения. Вывести полученную очередь на печать

[наверх](file:///D:\%D0%A1%D0%B8%D0%B1%D0%93%D1%83%D1%82%D0%B8\5%20%D1%81%D0%B5%D0%BC%D0%B5%D1%81%D1%82%D1%80\%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D1%8B%20%D0%B8%20%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D1%8B%20%D0%BE%D0%B1%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8%20%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85%20%D0%BD%D0%B0%20%D0%AD%D0%92%D0%9C\lr2.html#top)

[назад](file:///D:\%D0%A1%D0%B8%D0%B1%D0%93%D1%83%D1%82%D0%B8\5%20%D1%81%D0%B5%D0%BC%D0%B5%D1%81%D1%82%D1%80\%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D1%8B%20%D0%B8%20%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D1%8B%20%D0%BE%D0%B1%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8%20%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85%20%D0%BD%D0%B0%20%D0%AD%D0%92%D0%9C\lr1.html) | [содержание](file:///D:\%D0%A1%D0%B8%D0%B1%D0%93%D1%83%D1%82%D0%B8\5%20%D1%81%D0%B5%D0%BC%D0%B5%D1%81%D1%82%D1%80\%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D1%8B%20%D0%B8%20%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D1%8B%20%D0%BE%D0%B1%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8%20%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85%20%D0%BD%D0%B0%20%D0%AD%D0%92%D0%9C\lr_index.html) | [вперед](file:///D:\%D0%A1%D0%B8%D0%B1%D0%93%D1%83%D1%82%D0%B8\5%20%D1%81%D0%B5%D0%BC%D0%B5%D1%81%D1%82%D1%80\%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D1%8B%20%D0%B8%20%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D1%8B%20%D0%BE%D0%B1%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8%20%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85%20%D0%BD%D0%B0%20%D0%AD%D0%92%D0%9C\lr3.html)

**Лабораторная работа №3**

Динамические структуры данных. Списки.

Цель работы: Приобрести навыки создания односвязных списков и выполнения простейших операций над этими списками.

Подготовка к лабораторной работе

При подготовке к лабораторной работе следует повторить следующие вопросы: понятие списка, принцип организации односвязных списков, основные операции над списками, логическая и физическая структуры списков. В соответствии с заданием на лабораторную работу разработать схему алгоритма и составить программу на языке СИ.

Порядок выполнения работы

1. Разработать программу по заданию, выбранному в соответствии с вашим вариантом.
2. Ввести текст программы.
3. Произвести отладку программы.
4. Выполнить программу, провести анализ результатов и, убедившись в правильности решения.

Содержание отчета

Отчет по лабораторной работе должен содержать:

1. Задание к лабораторной работе.
2. Программу на языке СИ.
3. Результаты расчета.

Контрольные вопросы

1. Понятие связного списка.
2. Физическая и логическая структуры списка.
3. Основные операции над списками.
4. Дескриптор односвязного списка.

Задание к лабораторной работе

1. Дан список, содержащий информацию о 5 студентах: имена, фамилии, отчества и года рождения. Упорядочить их по году рождения и записать в новый список.
2. Создать список, содержащий вещественные числа. Вывести на экран содержимое списка, адрес первого и последнего элементов.
3. Создать список, содержащий сведения о группе студентов (фамилия, имя, отчество, пол студента, оценки по информатике). Ввести сведения еще об одном студенте и поместить их в список перед сведениями о первом студенте, фамилия которого начинается на букву «К».
4. Создать список, содержащий сведения о сотрудниках предприятия (фамилии, пол и стаж работы). Организовать еще два списка, в которые записать отдельно информацию о мужчинах и женщинах, размещая её в порядке возрастания стажа работы.
5. Даны года рождения и фамилии 10 студентов. Включить эту информацию в список в порядке убывания года рождения.
6. Список содержит произвольные целые числа. Удалить из списка все отрицательные элементы.
7. Создать список, содержащий фамилии студентов в алфавитном порядке. Удалить из списка фамилию студента, выбрав любой номер в журнале.
8. Дан стек, в который помещена информация о 5 студентах (имена фамилии, года рождения). Переписать эту информацию в список, упорядочив по году рождения.
9. Список содержит произвольные целые числа. Удалить из списка все положительные элементы, кратные трем.
10. Из элементов одного списка, содержащего сведения о студентах (фамилия, возраст) сформировать два других, записав в первый сведения о студентах моложе 20 лет, а во второй – старше 20 лет.
11. Список содержит произвольные целые числа. Удалить из списка все отрицательные нечетные элементы.
12. Из элементов списка, содержащего сведения об абонентах телефонной сети (фамилии, адреса, телефоны), создать список абонентов, номера телефонов которых начинаются с 266.
13. Список содержит произвольные целые числа. Исключить из данного списка каждый второй элемент, поместив их в другой список. Вывести на экран содержимое двух списков.
14. Список содержит фамилии студентов и их баллы в контрольный срок. Исключить из списка сведения о студентах, имеющих балл менее 1.
15. Список содержит произвольные целые числа. Удалить из списка все элементы, кратные пяти.
16. Список содержит произвольные целые числа. Удалить из списка все положительные элементы.
17. Создать произвольный список, содержащий целые числа. Исключить из списка четные числа и, если их останется меньше пяти, дополнить список нулями.
18. Дан список, содержащий целые числа. Написать программу, включающую в себя функции удаления первого элемента списка и вывода первого элемента списка на экран.
19. Список содержит произвольные целые числа. Удалить из списка все четные  элементы.
20. Список содержит произвольные целые числа. Удалить из списка все нечетные элементы.

[наверх](file:///D:\%D0%A1%D0%B8%D0%B1%D0%93%D1%83%D1%82%D0%B8\5%20%D1%81%D0%B5%D0%BC%D0%B5%D1%81%D1%82%D1%80\%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D1%8B%20%D0%B8%20%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D1%8B%20%D0%BE%D0%B1%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8%20%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85%20%D0%BD%D0%B0%20%D0%AD%D0%92%D0%9C\lr3.html#top)

[назад](file:///D:\%D0%A1%D0%B8%D0%B1%D0%93%D1%83%D1%82%D0%B8\5%20%D1%81%D0%B5%D0%BC%D0%B5%D1%81%D1%82%D1%80\%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D1%8B%20%D0%B8%20%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D1%8B%20%D0%BE%D0%B1%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8%20%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85%20%D0%BD%D0%B0%20%D0%AD%D0%92%D0%9C\lr2.html) | [содержание](file:///D:\%D0%A1%D0%B8%D0%B1%D0%93%D1%83%D1%82%D0%B8\5%20%D1%81%D0%B5%D0%BC%D0%B5%D1%81%D1%82%D1%80\%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D1%8B%20%D0%B8%20%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D1%8B%20%D0%BE%D0%B1%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8%20%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85%20%D0%BD%D0%B0%20%D0%AD%D0%92%D0%9C\lr_index.html)| [вперед](file:///D:\%D0%A1%D0%B8%D0%B1%D0%93%D1%83%D1%82%D0%B8\5%20%D1%81%D0%B5%D0%BC%D0%B5%D1%81%D1%82%D1%80\%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D1%8B%20%D0%B8%20%D0%B0%D0%BB%D0%B3%D0%BE%D1%80%D0%B8%D1%82%D0%BC%D1%8B%20%D0%BE%D0%B1%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8%20%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85%20%D0%BD%D0%B0%20%D0%AD%D0%92%D0%9C\lr4.html)

**Лабораторная работа №4**

Операции над бинарными деревьями.

Цель работы: Приобрести навыки работы с нелинейными списками.

Подготовка к лабораторной работе

При подготовке к лабораторной работе следует повторить следующие вопросы: понятие дерева, принципы организации дерева поиска, основные операции над деревьями , размещение древовидной структуры в памяти ЭВМ. В соответствии с заданием на лабораторную работу разработать схему алгоритма и составить программу на языке СИ.

Порядок выполнения работы

1. Разработать программу по заданию, выбранному в соответствии с вашим вариантом.
2. Создать исходный модуль с текстом программы.
3. Произвести отладку программы.
4. Выполнить программу на счет, провести анализ результатов и, убедившись в правильности решения.

Содержание отчета

Отчет по лабораторной работе должен содержать:

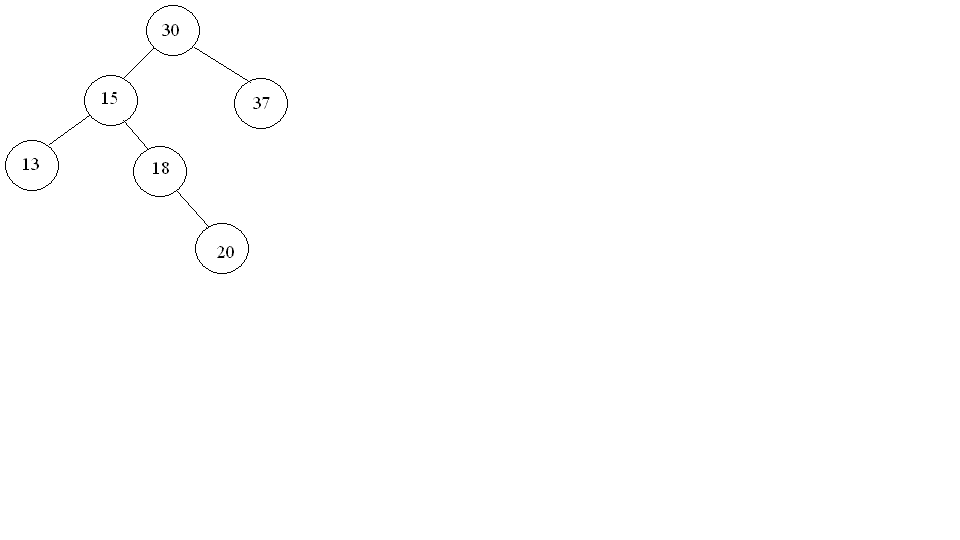
1. Задание к лабораторной работе.
2. Схема алгоритма решения задачи.
3. Листинг с текстом программы и результаты расчета.

Контрольные вопросы

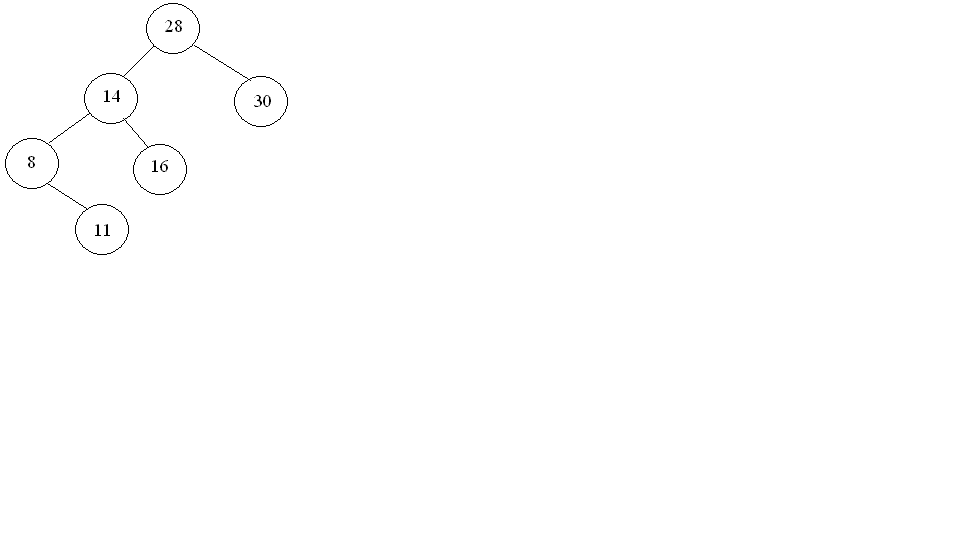
1. Понятие дерева.
2. Операции над бинарными деревьями.
3. Схемы обхода.
4. Прошивки деревьев.

Задание к лабораторной работе

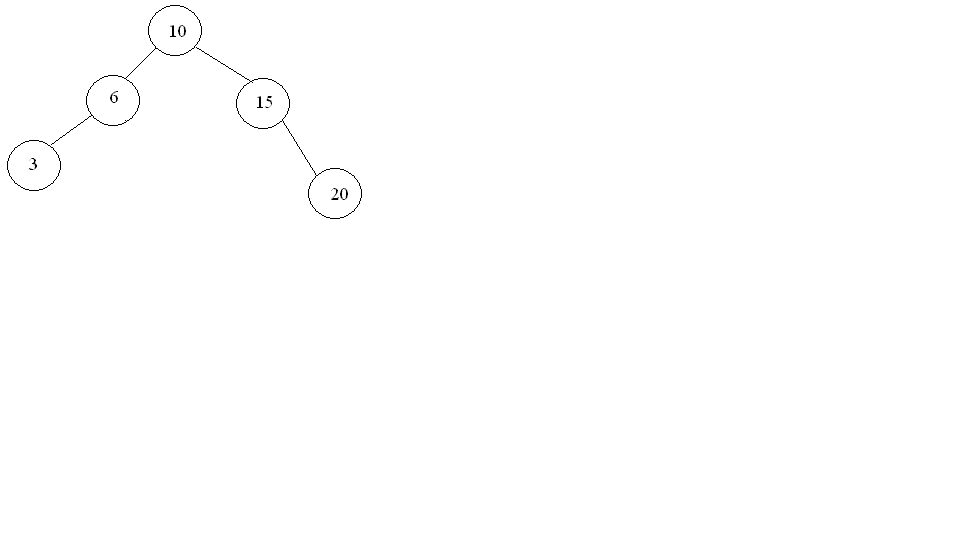
1. Построить бинарное дерево поиска. Подсчитать сумму значений всех ключей заданного бинарного дерева. Вывести узлы в порядке обхода слева.



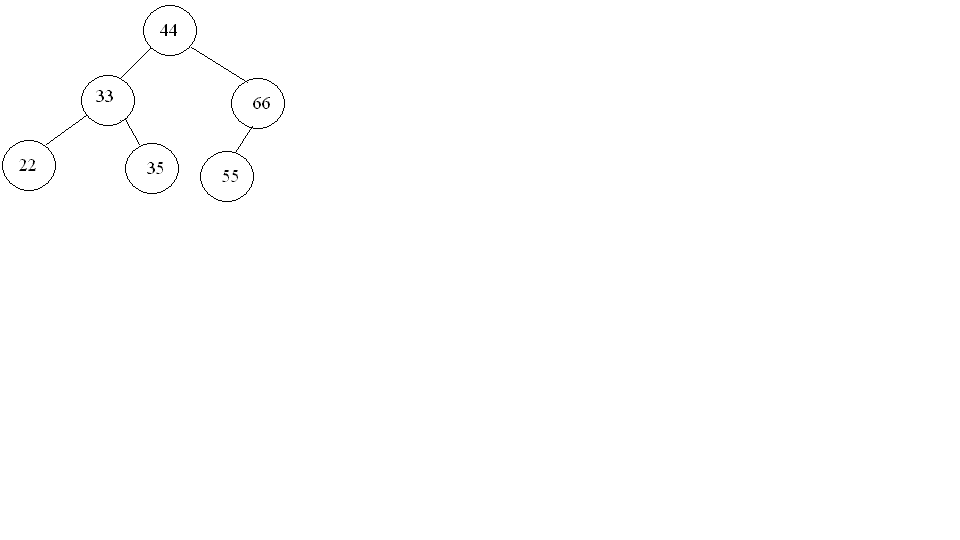
1. Построить бинарное дерево поиска. Подсчитать количество узлов с ключом меньше 15. Вывести узлы в порядке обхода сверху.



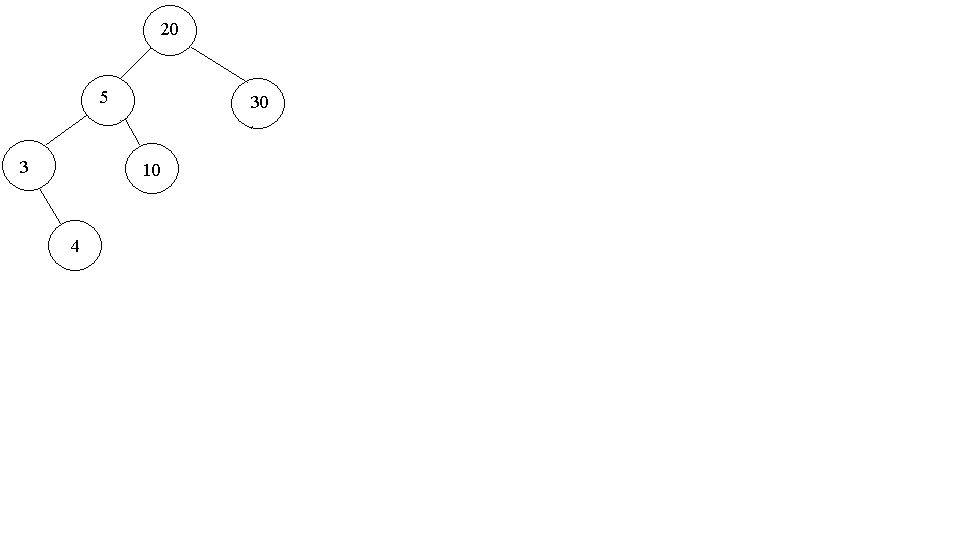
1. Построить бинарное дерево поиска. Удалить все узлы, имеющие значения ключей меньше 10. Вывести узлы в порядке обхода снизу.



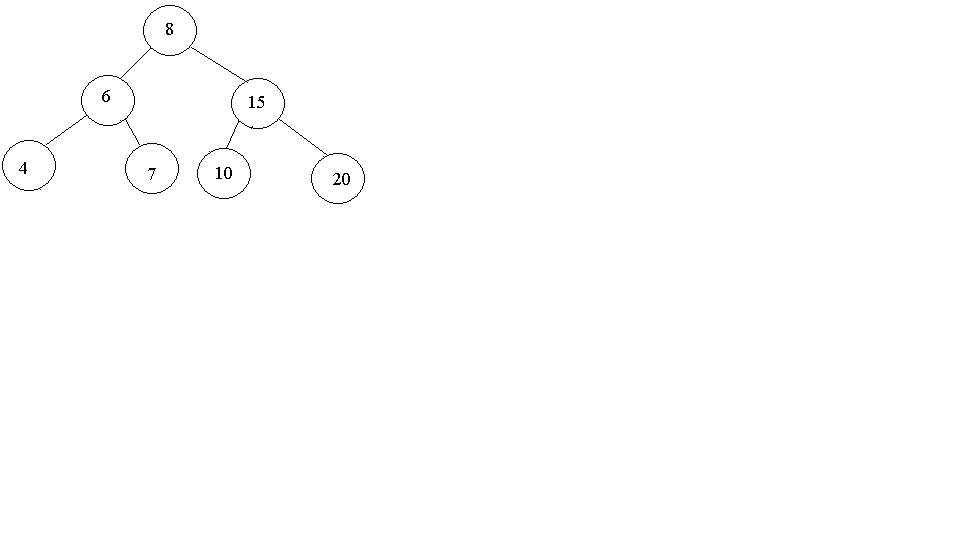
1. Построить бинарное дерево поиска. Определить значение ключа самого правого узла левого его поддерева. Вывести ключи узлов в порядке обхода снизу.



1. Построить бинарное дерево поиска. Подсчитать количество узлов с ключом больше 5. Вывести узлы в порядке обхода снизу.



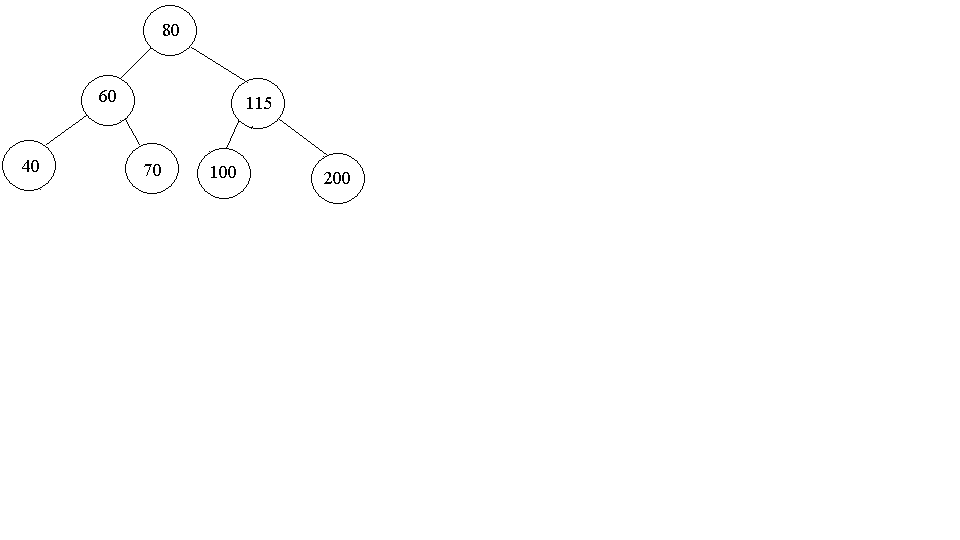
1. Дано бинарное дерево поиска. Исключить самый правый узел дерева. Вывести ключи узлов в порядке обхода сверху.



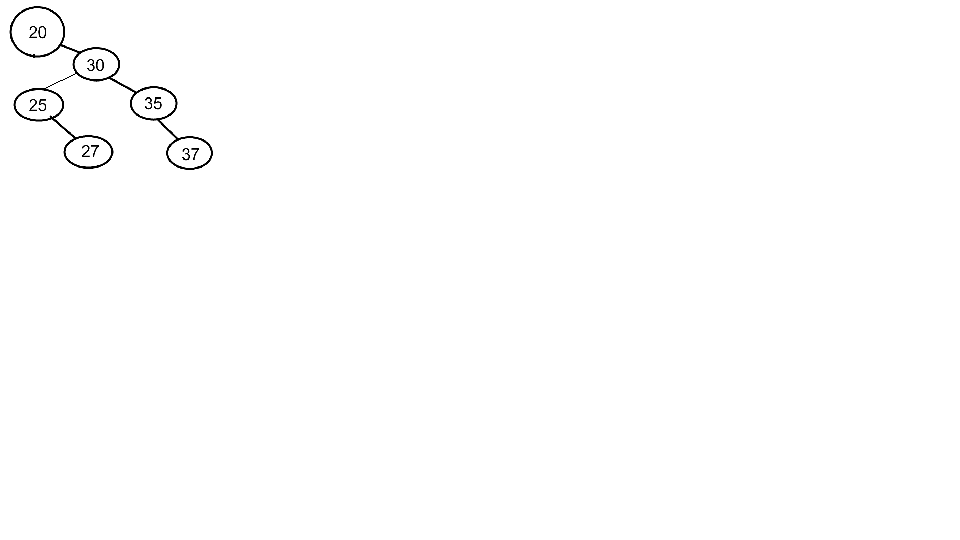
1. Построить бинарное дерево поиска. Подсчитать среднее арифметическое значение его узлов. Вывести узлы в порядке обхода слева.



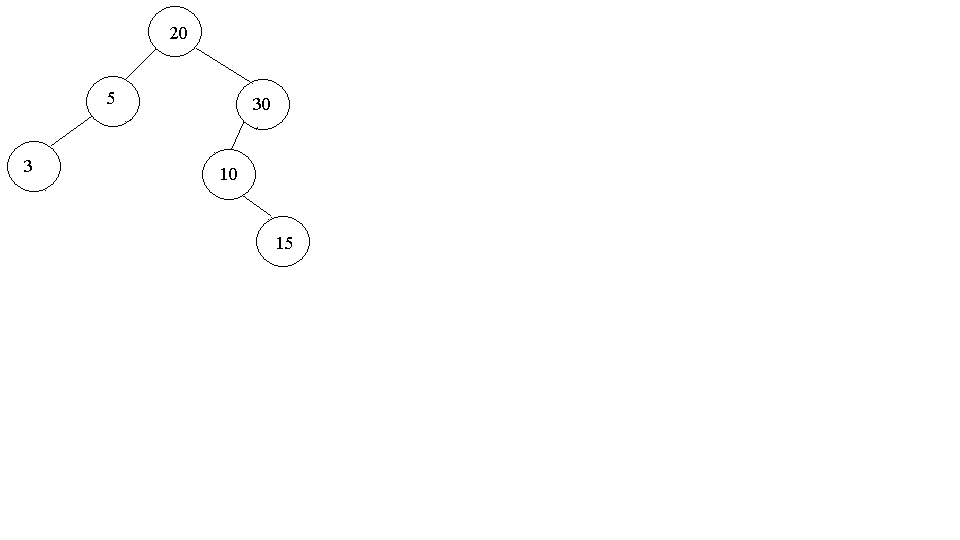
1. Дано бинарное дерево поиска. Определить значение ключа самого левого узла правого его поддерева. Вывести ключи узлов в порядке обхода сверху.



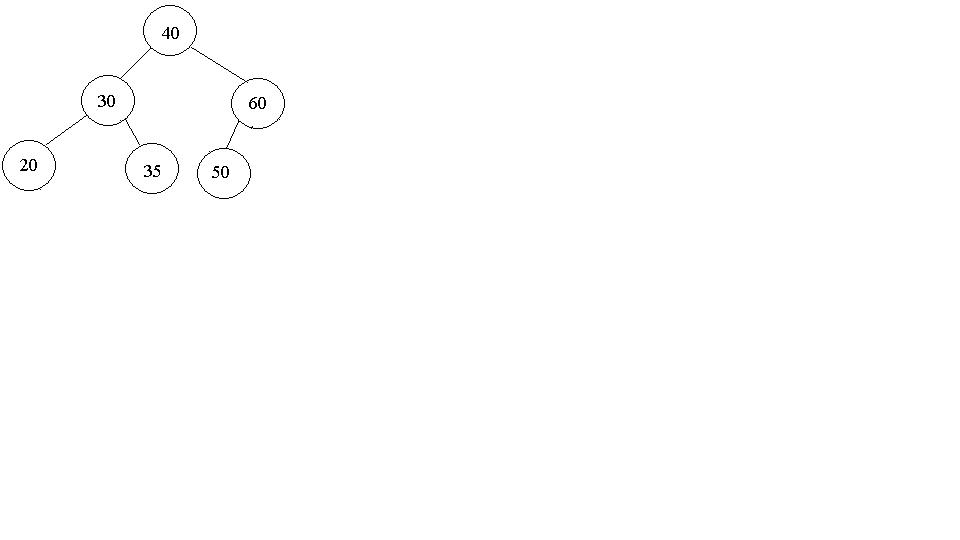
1. Построить бинарное дерево поиска. Подсчитать сумму ключей правого поддерева. Вывести ключи узлов в порядке обхода снизу.



1. Построить бинарное дерево поиска. Подсчитать сумму значений узлов, ключи которых не превышают 10. Вывести узлы в порядке обхода снизу.



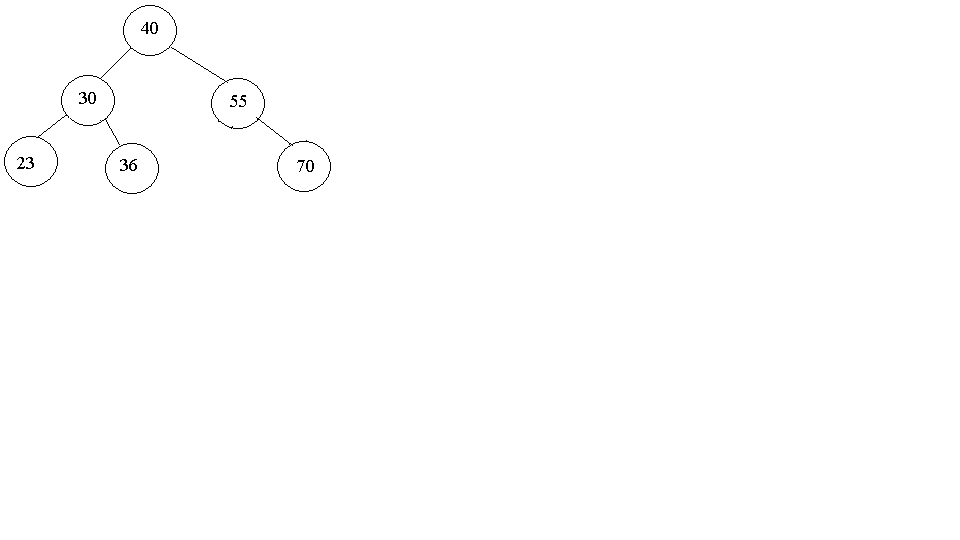
1. Построить бинарное дерево поиска. Подсчитать количество листьев. Вывести ключи узлов в порядке обхода снизу.



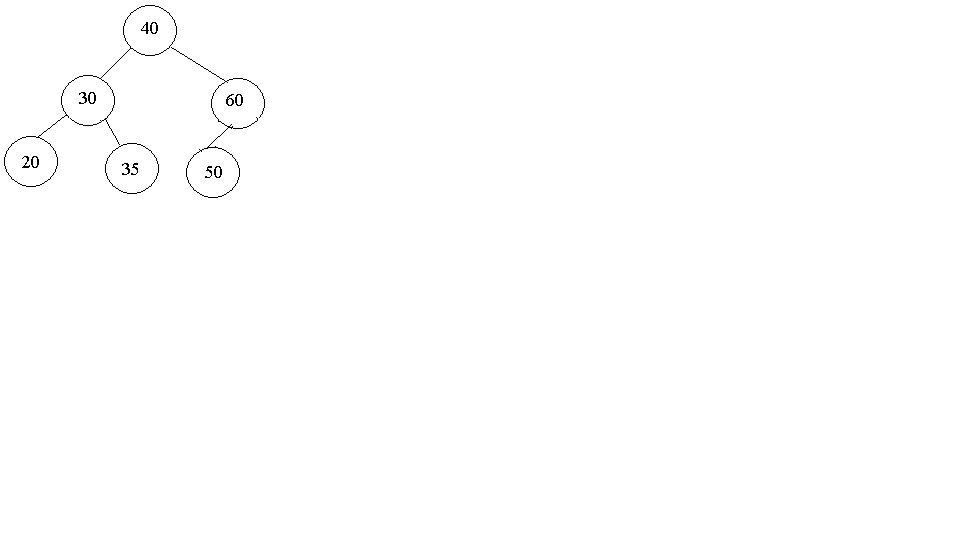
1. Дано бинарное дерево поиска. Подсчитать сумму значений узлов, являющихся листьями дерева. Вывести ключи узлов в порядке обхода сверху.



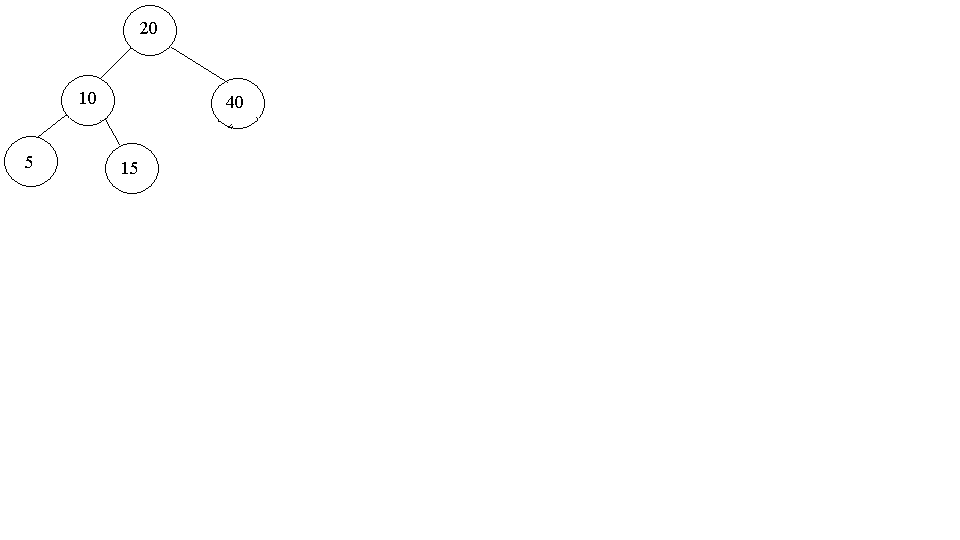
1. Построить бинарное дерево поиска. Удалить все узлы, имеющие значения ключей больше 30. Вывести ключи узлов в порядке обхода слева.



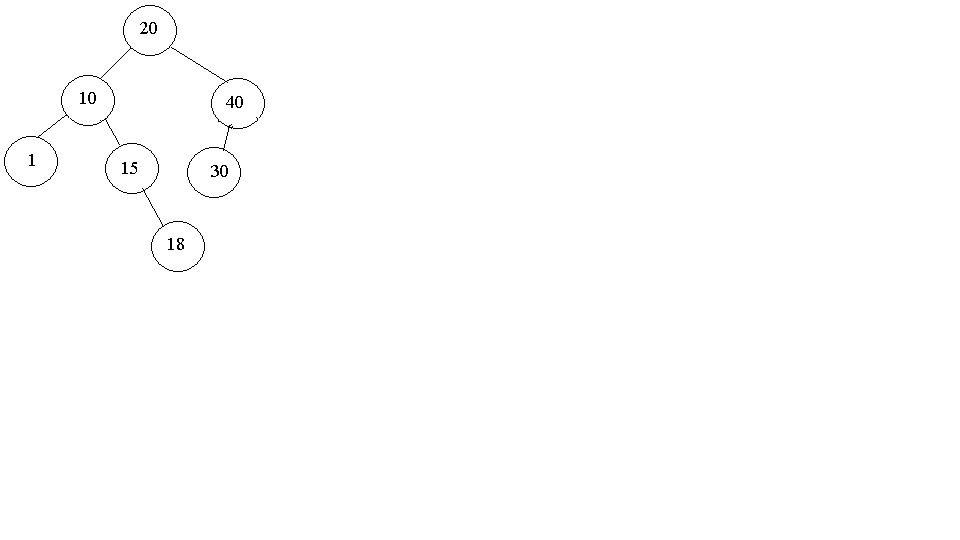
1. Построить бинарное дерево поиска. Определить значение ключа самого правого узла левого его поддерева. Вывести ключи узлов в порядке обхода снизу.



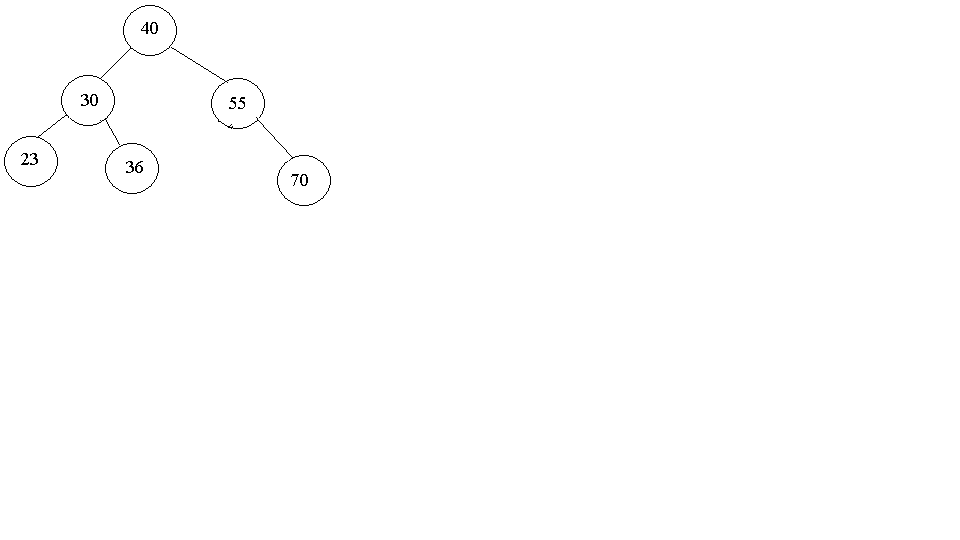
1. Дано бинарное дерево поиска. Найти максимальное значение ключа в заданном дереве поиска. Вывести ключи узлов в порядке обхода слева.



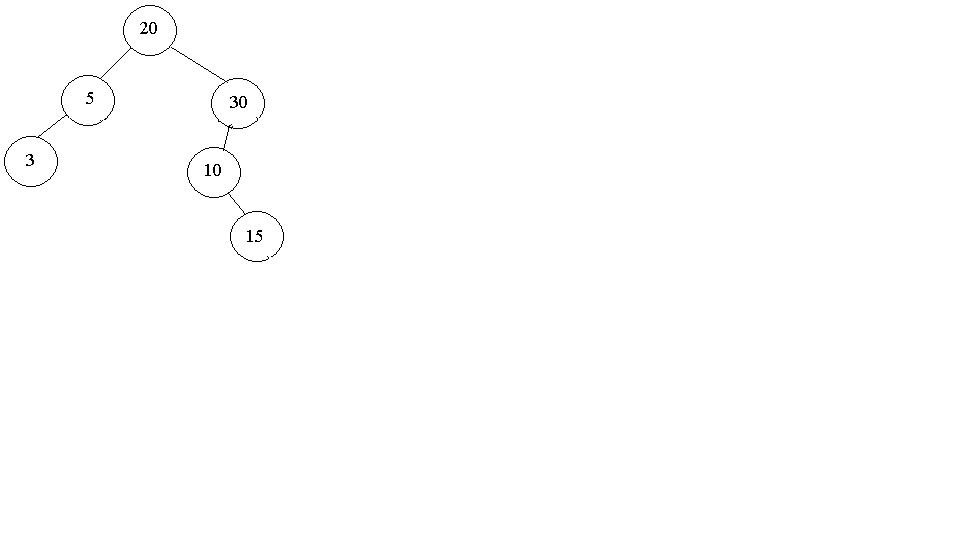
1. Дано бинарное дерево поиска. Найти минимальное значение ключа в заданном дереве поиска. Вывести ключи узлов в порядке обхода слева.



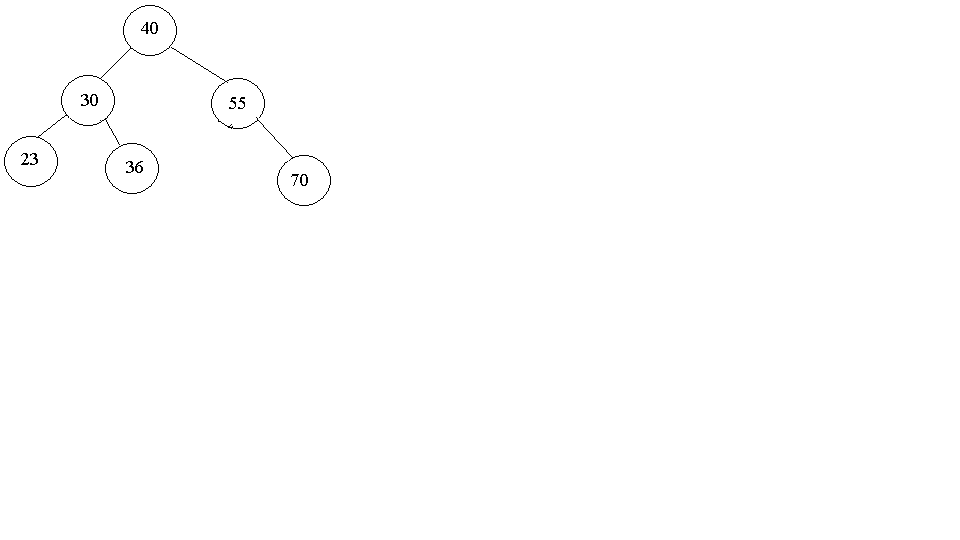
1. Построить бинарное дерево поиска. Удалить все узлы, имеющие значения ключей меньше 30. Вывести ключи узлов в порядке обхода слева.



1. Построить бинарное дерево поиска. Подсчитать сумму значений узлов, ключи которых не превышают 20. Вывести узлы в порядке обхода снизу.



1. Построить бинарное дерево поиска. Подсчитать сумму ключей левого поддерева. Вывести ключи узлов в порядке обхода снизу.



1. Построить бинарное дерево поиска. Подсчитать количество узлов в левом и правом поддеревьях отдельно. Вывести узлы в порядке обхода снизу.

