

Самостоятельная работа 9

Тема. Рекурсивные алгоритмы и их реализация

Цель:

- получить знания и практические навыки по разработке, рекурсивных алгоритмов;
- получить навыки реализации рекурсивных алгоритмов;
- получить умения и навыки в определении асимптотической сложности рекурсивного алгоритма при применении известных методов.

1. Требования к выполнению практической работы

В данной практической работе требуется реализовать два задания. Варианты задач к заданиям в табл. 20.

Первое задание направлено на освоении процесса разработки алгоритма, включающего: определение рекурсивного процесса решения задачи, запись его рекурсивной зависимости, оценка сложности рекурсивного алгоритма подходящим для него методом, реализация алгоритма. В этом задании решается первая задача варианта.

Второе задание направлено на закрепление знаний и умений по разработке и реализации более сложных рекурсивных процессов. В этом задании решается вторая задача варианта.

2. Задание 1

Реализовать рекурсивную функцию для решения первой задачи варианта.

2.1. Требования к выполнению задания 1

1. Выполнить разработку и оценить сложность алгоритма, для этого:

- определить, что процесс решения задачи может быть рекурсивным, для этого приведите рекуррентную зависимость процесса;
- определить глубину рекурсии, изменяя исходные данные;
- определить сложность рекурсивного алгоритма, используя метод подстановки и дерево рекурсии;
- проверить возможность применения основного метода оценки сложности рекурсивного алгоритма, если метод допустим, то проверить, что уже полученное вами решение совпадает с решением по этому методу.

2. Реализовать рекуррентную зависимость, разработав рекурсивную функцию.

Примечание. При разработке алгоритма принимайте во внимание результат алгоритма. Если алгоритм формирует одно значение простого или ссылочного типа, то его, возможно удобно реализовать как функцию, возвращающую результат.

3. Подготовить тесты.
4. Разработать программу, демонстрирующую получение решения задачи на подготовленных тестах.
5. Оформить отчет.

3. Задание 2

Разработать рекурсивную функцию (функции) для обработки списковой структуры согласно варианту.

3.1. Требования к выполнению задания 2

1. Разработать функцию создания исходного списка.

Для создания списка может быть разработана простая или рекурсивная функция по желанию (в тех вариантах, где не требуется рекурсивное создание списка).

2. Разработать рекурсивную функцию (функции), реализующую вторую задачу варианта.
3. Оформить отчет.

4. Варианты задач

Таблица 20. Варианты задач к заданиям 1 и 2 практической работы 8

№	Задачи варианта
1.	1. Найти наибольший общий делитель двух целых чисел. 2. Создание и вывод линейного однонаправленного списка из n элементов.
2.	1. Дан массив из n элементов целого типа. Найти минимальное значение массива. Считать, что оно одно. 2. В однонаправленном списке из n элементов найти элемент с заданным значением и вернуть на него указатель.
3.	1. Определить делится ли число на каждую из своих цифр. 2. В двунаправленном списке из n узлов, перевернуть часть списка, начиная с узла с номером p1 до узла с номером p2 ($p1 < p2$).
4.	1. Определить является ли текст – палиндромом. 2. Удалить из связного однонаправленного списка все элементы, равные заданному.
5.	1. Дан массив из n элементов вещественного типа. Вычислить среднее значение всех элементов массива. 2. Создание связного стека из n элементов.

№	Задачи варианта
6.	1. Сколько квадратов можно отрезать от прямоугольника со сторонами a и b . 2. Удаление стека, реализованного на однонаправленном списке.
7.	1. Найти максимальный элемент в массиве из n элементов. 2. Создание очереди на однонаправленном списке.
8.	1. Перевести число из 10-системы счисления в систему с основанием $B(1 < B < 10)$. Алгоритм выводит цифры кода в консоль. 2. Удаление очереди, реализованной на однонаправленном списке.
9.	1. Бинарный поиск элемента в массиве. 2. Создание двунаправленного списка.
10.	1. Вычислить значение цифрового корня для некоторого целого числа N . 2. Найти в двунаправленном списке количество четных элементов.
11.	1. Вычислить $x_1(x_2+x_3)(x_4+x_5+x_6) \dots (x_{46}+x_{47}+\dots+x_{55})$. 2. Удаление двунаправленного списка.
12.	1. Сортировка массива по возрастанию. 2. Создать новый однонаправленный список из исходного однонаправленного списка, являющийся перевертышем исходного.
13.	1. Дана последовательность из N чисел X_1, X_2, \dots, X_N . Вычислить значение выражения: $X_n(X_n+X_{n-1})(X_n+X_{n-1}+X_{n-2})(X_n+X_{n-1}+X_{n-2}+X_{n-3}) \dots (X_n+X_{n-1}+X_{n-2}+\dots+X_1)$. Массив не использовать. 2. Удалить из однонаправленного списка нули.
14.	1. Дана строка. Выполнить переворот строки (записать наоборот) на ее же месте в памяти. 2. Определить количество вхождений: положительных, отрицательных, нулевых значений в линейный однонаправленный список.
15.	1. Ханойская башня. 2. Удалить двунаправленный список.
16.	1. Прохождение лабиринта. 2. Определить симметрично ли число, цифры которого последовательно записаны в узлах двунаправленного списка.
17.	1. Проверить баланс скобок в арифметическом выражении, которое передано как строка. 2. Найти минимальное значение в однонаправленном списке.
18.	1. Дано целое число. Найти максимальную цифру десятичного числа. 2. Определить количество узлов однонаправленного списка, содержащих нечетное значение.

№	Задачи варианта
19.	<p>1. Дано целое десятичное число. Добавить цифру в начало числа.</p> <p>2. Даны два линейных однонаправленных списка. Определить, равны ли они. Должны совпадать по длине и по значениям узлов одинаковым номером.</p>
20.	<p>1. Дано целое число. Определить количество делителей числа.</p> <p>2. Дан линейный однонаправленный список. Сформировать новый список, записав него четные числа, но в обратном порядке, по отношению к их расположению в исходном.</p>
21.	<p>1. Вычислить с заданной точностью $\varepsilon > 0$ значение функции $y = e^x$, используя представление ее рядом. Точность считается достигнутой, если очередное слагаемое по модулю меньше ε. Гармоничный ряд вычисления:</p> $y = e^x = 1 + x/1! + x^2/2! + \dots + x^n/n! + \dots$ <p>2. Дан линейный однонаправленный список. Переформировать его так, чтобы он начинался с узла с заданным номером, а все узлы исходного списка до этого номера разместились после последнего узла и в обратном порядке.</p>
22.	<p>1. Вычислить с заданной точностью $\varepsilon > 0$ значение функции $y = \cos(x)$, используя представление ее рядом. Точность считается достигнутой, если очередное слагаемое по модулю меньше ε. Гармоничный ряд вычисления:</p> $y = \cos(x) = 1 - x^2/2! + x^4/4! - \dots + (-1)^n x^{2n}/(2n!) + \dots$ <p>2. Дан линейный однонаправленный список, значения узлов которого символы, и его длина. Длина списка четное число. Определить содержит ли список слово палиндром.</p>
23.	<p>1. Разработать рекурсивную функцию, которая реализует синтаксический анализатор для понятия идентификатор.</p> $identifier ::= \begin{cases} letter \\ identifier \begin{cases} digit \\ letter \end{cases} \end{cases}$ <p>2. Дан линейный однонаправленный список. Сформировать новый список из значений узлов исходного, записывая в него значения в обратном порядке.</p>
24.	<p>1. Вычислить количество комбинаций из n разных элементов по m. Количество комбинаций определяется формулой:</p> $C_n^m = \begin{cases} 1, & \text{если } m = 0, n > 0 \text{ або } m = n \geq 0; \\ 0, & \text{если } m > n \geq 0; \\ C_{n-1}^{m-1} + C_{n-1}^m & \text{иначе.} \end{cases}$ <p>2. Дан линейный однонаправленный список. Вывести значения узлов списка по правилу: сначала все отрицательные, затем все остальные.</p>

№	Задачи варианта
25.	<p>1. Дана последовательность натуральных чисел (одно число в строке), завершающаяся числом 0. Определите, какое количество элементов этой последовательности, равны ее наибольшему элементу.</p> <p>2. Дан линейный однонаправленный список, в каждом узле одна десятичная цифра некоторого десятичного числа. Сформировать десятичное число, обратное числу, хранящемуся в списке.</p>
26.	<p>1. Даны числа a и b. Определите, сколько существует последовательностей из a нулей и b единиц, в которых никакие два нуля не стоят рядом.</p> <p>2. Дан линейный однонаправленный список. Сформировать десятичное число из нечетных цифр списка, включая их в число в обратном порядке.</p>
27.	<p>1. Дана строка, содержащая только английские буквы (большие и маленькие). Добавить открывающиеся и закрывающиеся скобки по следующему образцу: example -> e(x(a(m)p)l)e . До середины добавляются открывающиеся скобки, после середины – закрывающиеся. В случае, когда длина строки четна, в скобках, расположенных в середине, должно быть 2 символа: card -> c(ar)d, но не c(a)r)d.</p> <p>2. Дан линейный однонаправленный список из n узлов ($n \geq 3$). Удалить последние три узла списка.</p>
28.	<p>1. Строки АВАСАВА формируются по следующему правилу. Первая строка — это просто "А" (без кавычек). Каждая последующая строка получается так: берется предыдущая строка, к ней приписывается первая буква, которая в ней не встречается, и потом еще раз предыдущая строка. Соответственно, вторая строка получается "А" + "В" + "А" = "АВА", третья строка — "АВА" + "С" + "АВА" = "АВАСАВА", четвертая — "АВАСАВА" + "D" + "АВАСАВА" = "АВАСАВАДАВАСАВА" и т.д. Сформируйте по этому правилу n-ую строку ($1 \leq n \leq 6$).</p> <p>2. Дан линейный однонаправленный список, узлы которого целые числа. Сформировать новый список обратный исходному.</p>
29.	<p>1. Организовать ввод-вывод последовательности n целых чисел, с применением рекурсии. Числа должны выводиться с порядковыми номерами и в том же порядке, в каком производился их ввод. Использование массивов не разрешается.</p> <p>2. Дан линейный однонаправленный список. Сформировать новый список из нечетных значений исходного. Порядок следования значений в узлах нового списка соответствует порядку следования значений в исходном списке.</p>

№	Задачи варианта
30.	<p>1. Составить программу, которая, используя рекурсивную функцию, находит значение данной функции для любых целых неотрицательных аргументов n и a:</p> $R(n, a) = \sqrt{a + \underbrace{\sqrt{a + \dots + \sqrt{a}}}_{n\text{-корней}}}$ <p>2. Дан линейный однонаправленный список. Реализовать функцию, реализующую удаление узла из списка по алгоритму клавиши BackSpace.</p>
31.	<p>1. Определение корней уравнения (например, такого $y=x^2-2$) на заданном отрезке методом половинного деления с заданной точностью $\varepsilon > 0$. Для простоты будем считать, что отрезок задается таким образом, что корень на нем есть (иначе основная программа должна содержать проверку наличия корня).</p> <p><i>Метод половинного деления.</i> Корень расположен между серединой отрезка и тем концом, значение функции в котором по знаку не совпадает со значением функции в середине отрезка. Если абсолютная величина функции в середине отрезка не превышает заданного значения точности, то координата середины отрезка и есть корень.</p>

5. Структура отчета

Титульный лист.

Оглавление.

1. Задание 1.

1.1. Условие задания и задачи варианта.

1.2. Описание алгоритма (представить рекуррентную зависимость).

1.3. Выполнение анализа сложности алгоритма.

1.3.1. Приведите рекуррентное соотношение для алгоритма.

1.3.2. Приведите процесс получения решения рекуррентного соотношения методами подстановки и дерево рекурсии.

1.3.3. Обоснуйте применение основной теоремы для получения (или не получения) решения рекуррентного соотношения. Если применима теорема для представленного соотношения, то опишите процесс получения решения.

1.4. Тесты.

1.5. Коды используемых функций.

1.6. Код программы и скриншоты результатов тестирования.

2. Задание 2.

2.1. Условие задания и задачи варианта.

- 2.2. Тесты.
- 2.3. Коды используемых функций.
- 2.4. Код программы и скриншоты результатов тестирования.
3. Привести выводы по полученным знаниям и умениям.
4. Список информационных источников, которые были использованы при выполнении задания.

6. Примеры реализации рекурсивных алгоритмов

Задача 1. Дана последовательность целых чисел, заканчивающаяся нулем. Вывести сначала положительные, а затем отрицательные значения.

Рекурсивная зависимость, представляющая алгоритм.

Такое определение алгоритма говорит об его рекурсивной природе.

$$rec1() = \begin{cases} \text{Ввод в переменную } n \text{ числа последовательности} \\ \text{Вывод } n \text{ и шаг в рекурсию при } n > 0 \\ \text{Шаг в рекурсию и вывод } n \text{ при } n < 0 \\ \text{Выход из рекурсии при } n = 0 \end{cases}$$

Задача 2. Вычислить x^n . При $x=0$ и $n<0$ результат INFINITY.

Рекурсивная зависимость, представляющая алгоритм.

$$rec2(x, n) = \begin{cases} 1 & \text{если } n = 0 \\ x * rec2(x, n - 1) & \text{если } n > 0 \\ 1/rec2(x, |n|) & \text{если } n < 0 \end{cases}$$

Коды рекурсивных алгоритмов.

<pre> void rec1() { int n; cin>>n; if (n==0) return; else if(n>0) { cout<<n; rec1(); } else </pre>	<pre> double rec2(int x, int n) { if (n==0) { return 1; } if (n>0) // step recursii rec2(x,n)=x*rec2(x,n-1) return x*rec2(x,n-1); if(n<0){ return 1/rec2(x,abs(n)); } } </pre>
---	---

<pre> { rec11(); cout<<n; } </pre>	<pre> } </pre>
--	----------------

7. Контрольные вопросы

1. Какой алгоритм называют рекурсивным?
2. Что определяет термин *Шаг рекурсии*?
3. Что определяет термин *Глубина рекурсии*?
4. Когда наступает завершение рекурсии?
5. Какая рекурсия называется прямой рекурсией?
6. Как организуется косвенная рекурсия?
7. Как реализуется алгоритм прямой линейной рекурсии?
8. Как по коду функции определить, что функция реализует каскадную рекурсию?
9. Что хранит элемент стека при каждом вызове рекурсивной функции?
10. Что происходит со значениями элементов стека при завершении выполнения рекурсивной функции?
11. Изобразите модель стека рекурсивных вызовов для алгоритма вычисления $n!$.
12. Изобразите дерево рекурсии вычисления значения $5!$ и пятого числа Фибоначчи.
13. Как по дереву рекурсии алгоритма можно определить глубину рекурсии?
14. Как связана глубина рекурсии и стек рекурсивных вызовов одного алгоритма?
15. Как получить асимптотическую оценку сложности рекурсивного алгоритма?
16. Как называется метод разработки алгоритма, рекуррентное соотношение которого для задачи размера n записано так:
 $T(n) = aT(n/b) + c$.
17. Даны два рекуррентных соотношения:
 $T(n) = 2T(n/2) + \theta(n)$ и $T(n) = T(n-1) + c$
 К какому из них можно применить основной метод для получения решения?
18. Определите глубину рекурсии алгоритма при значениях: $a=7, b=2$.

```

int count (int a, int b) //a>0, b>0 {
    if(a==b) return 1;
    else

```



```
if(a>b) return 1+ count(a-b,b);  
else return 1+count(a,b-a);  
}
```

19. Добавьте в строку 2 оператор, завершающий рекурсию алгоритма.

```
1. int count (int a, int b) //a>0, b>0 {  
2. if(a==b) _____;  
3. else  
4. if(a>b) return 1+ count(a-b,b);  
5. else return 1+count(a,b-a);  
6. }
```

20. Дано рекуррентное соотношение времени выполнения рекурсивного алгоритма. Найти решение рекуррентного соотношения.

$$T(n) = \begin{cases} \Theta(1) & \text{при } n = 1 \\ 2T(n/2) + \Theta(n) & \text{при } n > 1 \end{cases}$$