**Практическое задание 6**

**Тема.** Рекурсивные алгоритмы и их реализация.

**Цель.** Получить знания и практические навыки по разработке и реализации рекурсивных процессов.

**Задание.**

Разработать и протестировать рекурсивные функции в соответствии с задачами варианта. Составить отчет.

Требования к выполнению первой задачи варианта:

* приведите итерационный алгоритм решения задачи
* реализуйте алгоритм в виде функции и отладьте его
* определите теоретическую сложность алгоритма
* реализуйте и отладьте рекурсивную функцию решения задачи
* определите глубину рекурсии, изменяя исходные данные
* определите сложность рекурсивного алгоритма, используя метод подстановки и дерево рекурсии
* приведите для одного из значений схему рекурсивных вызовов
* разработайте программу, демонстрирующую выполнение обеих функций, и покажите результаты тестирования.

Требования к выполнению второй задачи варианта:

* рекурсивную функцию для обработки списковой структуры согласно варианту. Информационная часть узла – простого типа – целого;
* определите глубину рекурсии
* определите теоретическую сложность алгоритма
* разработайте программу, демонстрирующую работу функций и покажите результаты тестов.

Таблица 1. Варианты заданий

|  |  |
| --- | --- |
| № | 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 (р1<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. |
| 6 | 1. Сколько квадратов можно отрезать от прямоугольника со сторонами a и в. 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 В(1<В<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 x1(x2+x3)(x4+x5+x6)....(x46+x47+...+x55). 2. Удаление двунаправленного списка |
| 12 | 1. Сортировка массива по возрастанию 2. Создать новый однонаправленный список из исходного однонаправленного списка, записав его элементы наоборот. |
| 13 | 1. Дана последовательность из N чисел Х1,Х2,....,ХN. Вычислить значение выражения: Хn(Хn+Xn-1)(Хn+Xn-1+Xn-2)(Хn+Xn-1+Xn-2+Xn-3)... (Хn+Xn-1+Xn-2+...+X1). 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. |
| 19 | 1. Дано целое десятичное число. Добавить цифру в начало числа. 2. Даны два линейных однонаправленных списка. Определить, равны ли они. Должны совпадать по длине и по значениям узлов одинаковым номером. |
| 20 | 1. Дано целое число. Определить количество делителей числа. 2. Дан линейный однонаправленный список. Сформировать новый список, записав него четные числа, но в обратном порядке, по отношению к их расположению в исходном. |
| 21 | 1. Вычислить с заданной точностью ε>0 значение функции y=ex, используя представление ее рядом. Точность считается достигнутой, если очередное слагаемое по модулю меньше ε. Гармоничный ряд вычисления   y=ex=1+x/1!+x2/2!+…+ xn/n!+…..   1. Дан линейный однонаправленный список. Переформировать его так, чтобы он начинался с узда с заданным номером, а все узлы исходного списка до этого номера разместились после последнего узла и в обратном порядке. |
| 22 | 1. Вычислить с заданной точностью ε>0 значение функции y=cos(x), используя представление ее рядом. Точность считается достигнутой, если очередное слагаемое по модулю меньше ε. Гармоничный ряд вычисления   y=cos(x)=1-x2/2!+x4/4!-…+(-1)nx2n/(2n!)+….   1. Дан линейный однонаправленный список, значения узлов которого символы, и его длина. Длина списка четное число. Определить содержит ли список слово палиндром. |
| 23 | 1. Разработать рекурсивную функцию, которая реализует синтаксический анализатор для понятия идентификатор.   https://www.bestprog.net/wp-content/uploads/2019/01/recursion_04_-300x111.jpg   1. Дан линейный однонаправленный список. Сформировать новый список из значений узлов исходного, занеся в него значения в обратном порядке |
| 24 | 1. Вычислить количество комбинаций из n разных элементов по m. Количество комбинаций определяется формулой   https://www.bestprog.net/wp-content/uploads/2019/01/recursion_03r-300x89.jpg   1. Дан линейный однонаправленный список. Вывести значения узлов списка по правилу: сначала все отрицательные, затем все остальные. |
| 25 | 1. Дана последовательность натуральных чисел (одно число в строке), завершающаяся числом 0. Определите, какое количество элементов этой последовательности, равны ее наибольшему элементу. 2. Дан линейный однонаправленный список, в каждом узле одна десятичная цифра некоторого десятичного числа. Сформировать десятичное число, обратное числу, хранящемуся в списке. |
| 26 | 1. Даны числа a и b. Определите, сколько существует последовательностей из a нулей и b единиц, в которых никакие два нуля не стоят рядом. 2. Дан линейный однонаправленный список. Сформировать десятичное число из нечетных цифр списка, включая их в число в обратном порядке. |
| 27 | 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.   1. Дан линейный однонаправленный список из n узлов (n>=3). Удалить последние три узла списка. |
| 28 | 1. Строки ABACABA формируются по следующему правилу. Первая строка — это просто "A" (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 "A" + "B" + "A" = "ABA", третья строка — "ABA" + "C" + "ABA" = "ABACABA", четвертая — "ABACABA" + "D" + "ABACABA" = "ABACABADABACABA" и т.д.   Сформируйте по этому правилу n-ую строку (1<=n<=6).   1. Дан линейный однонаправленный список, узлы которого целые числа. Сформировать новый список обратный исходному |
| 29 | 1. Организовать ввод-вывод последовательности n целых чисел, с применением рекурсии. Числа должны выводится с порядковыми номерами и в том же порядке, в каком производился их ввод. Использование массивов не разрешается. 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. |
| 30 | 1. Составить программу, которая, используя рекурсивную функцию, находит значение данной функции для любых целых неотрицательных аргументов n и a:      1. Дан линейный однонаправленный список. Реализовать функцию, реализующую удаление узла из списка по алгоритму клавиши BackSpace. |
| 31 | 1. Определение корней уравнения (например, такого y=x2-2 ) на заданном отрезке методом половинного деления с заданной точностью ε>0/. Для простоты будем считать, что отрезок задается таким образом, что корень на нем есть (иначе основная программа должна содержать проверку наличия корня).   *Метод половинного деления*. 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. |

Вопросы по теме

1. Определение рекурсивной функции
2. Шаг рекурсии
3. Глубина рекурсии
4. Условие завершения рекурсии
5. Виды рекурсии
   1. - линейная -
   2. - каскадная –
6. Прямая и косвенная рекурсия
7. Организация стека рекурсивных вызовов

**Примеры реализации рекурсивных алгоритмов**

#include "stdafx.h"

#include "iostream.h"

// Задача 1.Дана последовательность целых чисел, заканчивающаяся нулем

// вывести сначала положительные, а затем отрицательные значения

void rec1();

Рекуррентная зависимость

//Задача 2. Вычислить xn. При x=0 и n<0 результат INFINITY

Такое определение алгоритма говорит об его рекурсивной природе

int rec2(int x, int n);

int main()

{

rec1();

cout<<rec2(2,3);

double rez=rec2(0, -3);

if (rez == INFINITY)

std::cout << "zero divide";

else

std::cout << rez;

return 0;

}

void rec1()

{int n;

cin>>n;

if (n==0)

return;

else

if(n>0)

{

cout<<n;

rec11();

}

else

{rec11();

cout<<n;

}

}

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));

}