Решить задачи на C++ , работа должна содержать пояснения к решению.

В задаче 1 предлагается освоить использование рекурсивных процедур. В задаче 2  создать динамическую структуру данных типа список.

# ЗАДАЧА 1

На следующей странице на рис. 1 изображено дерево с пятнадцатью вершинами 1,2,3, … 14, 15. Корнем дерева служит точка 1. Кроме того, для каждой вершины указана целочисленная пометка, согласно следующей ниже таблицы.

Таблица 1

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Номер вершины | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| Число-пометка | 10 | 22 | 16 | 11 | 45 | 25 | 25 | 4 | 10 | 7 | 8 | 25 | 10 | 1 | 9 |

Требуется выполнить задание 0 варианта, которое использует вышеприведѐнную таблицу 1 и топологическую структуру дерева на рис.1 и должно опираться на применение рекурсивных процедур.

Вариант 0.

Используя рекурсивные процедуры для перебора всех вершин дерева, определить сумму чисел-пометок всех его вершин.

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

Рис. 1

Пример.

С помощью рекурсивной процедуры пройти все вершины дерева с рис.1 и напечатать номера всех этих вершин.

Указание. Обратите внимание и воспользуйтесь тем обстоятельством, что вершины 8, 9, 10, … , 14, 15 не имеют потомков, а для любой другой вершины с номером n еѐ ближайшие потомки имеют номера 2\*n и 2\*n+1.

Решение.

Задача решается с помощью следующей программы, использующей рекурсивную процедуру.

program DemPrimer1;

{Рекурсивная процедура OchShag1(), позволяющая пройти и последовательно напечатать номера всех вершин дерева рис.1}

Procedure OchShag1(n:integer);

begin

if n<=7 then {вершина имеет двух потомков} begin

write(n:4); OchShag1(2\*n); OchShag1(2\*n+1); end; end; begin

{Выполняем этот перебор и печать} OchShag1(1); writeln; end.

Или же можно чуть усовершенствовать рекурсивную процедуру и тогда получится такой листинг.

program DemPrimer2;

{Рекурсивная процедура OchShag2(), позволяющая пройти и последовательно напечатать номера всех вершин дерева рис.1}

Procedure OchShag2(n:integer); var i : integer; begin write(n:4); {вершина имеет двух потомков} if n<=7 then

for i:=1 to 2 do OchShag2(2\*n+i-1); end;

begin

{Выполняем этот перебор и печать} OchShag2(1); writeln; end.

В результате выполнения этой программы напечатается следующая последовательность номеров вершин, -

1 2 4 8 9 5 10 11 3 6 12 13 7 14 15

# ЗАДАЧА 2

Записать в стек 5 целых чисел. Затем изменить порядок следования элементов в стеке согласно 0 варианту, меняя только адресную часть записи стека. Считать, что адресная часть первого элемента указывает на отсутствие следующего элемента списка. После чего вывести на печать содержимое стека (в обратном порядке), освобождая занятую память.

Вариант 0.

Поменять местами 4-й с 5-м элементом в стеке, не изменяя расположение в динамической памяти информационной части всех элементов стека.

Указание. Представим исходное состояние стека в следующем виде :

5

4

3

2

1

Nil

Затем, согласно, например, варианту 1, порядок следования изменится следующим образом (Вариант 1.Используя условие варианта 0, поменять местами 3-й с 5-м элементом в стеке.):

5

4

3

2

1

Nil

# Пример

Составить программу для решения задачи с тем же условием, что и в варианте 1, но список формируется из 6 чисел.

Далее следует листинг программы и выходных результатов.

Program DemPrim0;

Const NN=6;

Type

Uk=^Stek;

Stek = Record

I : Integer;

A : Uk

End;

Var

U1, U2 : Uk;

I1, J, J1 : Integer ;

A3, A4, A5, A6, UU : Uk;

Begin

U2 := Nil; I1 := 0; Writeln;

for J:=1 to NN do begin

New(U1); Write('Введите число : '); Readln(I1);

U1^.I := I1; U1^.A := U2; U2 := U1; end;

{Переставим в стеке значения третьего и пятого элементов} {Запомним значения указателей в соответств. элементах }

UU := U1; for J:=1 to NN do

begin

U2:= U1^.A; J1:=NN-J+1; {Номер элемента} If J1=3 then A3:=U2 else if J1=4 then A4:=U2 else if J1 = 5 then A5:=U2 else if J1=6 then A6:=U2;

U1:=U2; end;

{ Поменяем значения указателей должным образом}

U1:=UU;

For J:=1 to NN do

begin

U2:=U1^.A; J1:=NN-J+1; if J1=3 then U1^.A := A5 else if J1=4 then U1^.A:=A6 Else if J1=5 then U1^.A:=A3 else if J1=6 then U1^.A:=A4;

U1:=U2; end;

Writeln; U1:=UU;

Repeat

Writeln('Элемент стека - ',U1^.I);

U2:=U1^.A; Dispose(U1); U1:=U2; Until U1=Nil End.

Распечатка выходных результатов

Введите число : 1

Введите число : 2

Введите число : 3

Введите число : 4

Введите число : 5

Введите число : 6

Значение стека - 6

Значение стека - 3

Значение стека - 4

Значение стека - 5

Значение стека - 2

Значение стека - 1