УВАЖАЕМЫЕ СТУДЕНТЫ!

ВАМ НЕОБХОДИМО ВЫПОЛНИТЬ СЛЕДУЮЩЕЕ:

1. Ознакомиться с теорией, составить и ответить на вопросы.
2. Законспектируйте лекцию.
3. Отправить преподавателю на почту **v.vika2014@mail.ru** и указать свою Ф.И.О, группу, и название дисциплины тел 072-17-44-9-22

Тема: Рекурсивные методы построения алгоритмов

Рекурсия – способ организации вычислительного процесса, при котором функция в ходе выполнения составляющих ее операторов обращается сама к себе.

Использование рекурсии сокращает программу, в этом ее преимущество перед использованием итерационных циклов, но при выполнении может вызвать переполнение оперативной памяти компьютера.

Вид рекурсивной процедуры:

```
Procedure Rec (t:Integer);
Begin
<действия на вход в рекурсию>; -
if <Проверка условий> Then Rec (t-1);
<действия на выходе из рекурсии>
End;
```

Рассмотрим типовой пример задачи вычисления факториала числа.

```
Function Factorial ( n: Integer) : Longint;
Begin
  If n=1 Then Factorial:=1
         Else Factorial:=n* Factorial (n-1);
End;
```

На рис. 1 показаны прямой и обратный ход рекурсии. 1 В любой рекурсивной подпрограмме обязательно должна быть не рекурсивная ветвь:

```
If n = 1 Then Factorial := 1
```

Пусть переменной *a* присваивается значение 5!:

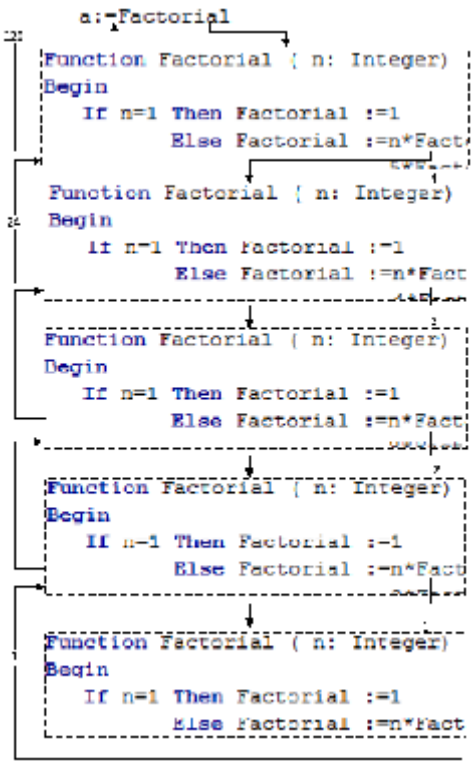


Рисунок 1 Прямой и обратный ход рекурсии.

Рассмотрим некоторые примеры использования рекурсии

1. Сложение двух чисел $a + b$. Рекурсивное определение операции сложения двух чисел:

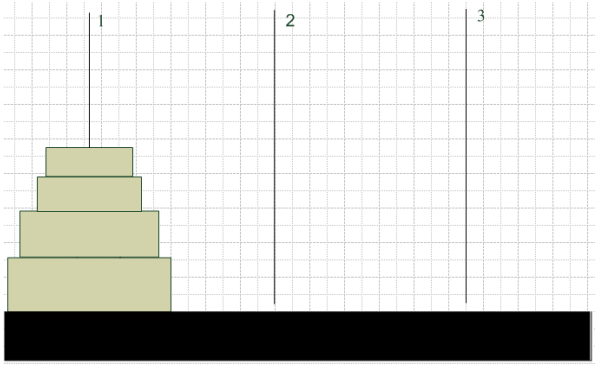
$$a + b = \begin{cases} a \text{ при } b=0, \\ (a+1)+(b-1) \text{ при } b>0 \\ (a-1)+(b+1) \text{ при } b<0 \end{cases}$$

```
Function Sum ( a, b: Integer) : Integer;
 Begin
  If b=0 Then Sum:=a
         Else If b>0 Then Sum:=Sum (a+1, b-1)
                     Else Sum:=Sum (a-1, b+1)
 End;
```

2. Каждое число Фибоначчи равно сумме двух предыдущих чисел при условии, что первые два равны 1 (1, 1, 2, 3, 5, 8, 13, 21,…).В общем виде n-е число Фибоначчи можно определить так: $\Phi(n)=\begin{cases} 1 \text{ при } n=1 \text{ или } n=2, \\ \Phi(n-1)+\Phi(n-2) \text{ при } n>2. \end{cases}$

```
Function Fib (n:Integer) :Integer;
 Begin
  If n<=2 Then Fib:=1
          Else Fib:=Fib (n-1) +Fib (n-2) ;
 End;
```

3. «Ханойские башни». В конце XIX века в Европе появилась игра под названием «Ханойские башни». Реквизит игры состоит из 3 игл, на которых размещается башня из колец. Цель игры – перенести башню с левой иглы (1) на правую (3), причем за один раз можно перенести только одно кольцо. Кроме того, запрещается помещать большее кольцо над меньшим.



Программа, которая печатает последовательность переноса колец посредством рекурсивной процедуры, представлена в Приложении 1, Hanoy.pas.

Данная задача своим происхождением обязана легенде, в которой рассказывается, что в большом храме Бенареса на бронзовой плите установлены три алмазных стержня, на один из которых бог нанизал во время сотворение мира 64 золотых диска. С тех пор день и ночь монахи, сменяя друг друга каждую секунду, перекладывают по одному диску согласно описанным выше правилам. Конец мира наступит тогда, когда все 64 диска будут перемещены, на что потребуется чуть больше 58 млрд. лет.