**Các bài tập liên quan đến giải đệ quy**

## Tính tổng dãy số Fibonacci đến số n

#include <stdio.h>

int F(int n){

if(n == 0 || n == 1){

return n;

}

else{

return F(n - 1) + F(n - 2);

}

}

int main(){

printf("%d", F(10));

return 0;

}

## Tính giai thừa n!

Code :

#include <stdio.h>

long long factorial(int n){

if(n == 0){

return 1;

}

else{

return n \* factorial(n - 1);

}

}

int main(){

printf("%lld", factorial(10));

return 0;

}

* Tìm chữ số lớn nhất của số tự nhiên n

Code :

#include <stdio.h>

int F(long long n){

if(n < 10){

return n;

}

else{

int tmp = F(n / 10);

return n % 10 > tmp ? n % 10 : tmp;

}

}

int main(){

long long n = 12349567;

printf("%d", F(n));

return 0;

}

## Tính tổng của các chữ số của số tự nhiên n

Code :

#include <stdio.h>

int sum(int n){

if(n < 10){

return n;

}

else{

return n % 10 + sum(n / 10);

}

}

int main(){

printf("%d", sum(12345));

return 0;

}

## Bài toán Tháp Hà Nội (Tower of Hanoi)

* Bài toán con ếch
* Tính S(n) = \sqrt {n + \sqrt {n - 1 + \sqrt {n - 2 + ... + \sqrt {2 + \sqrt 1 } } } }  (n dấu căn)
* Cho số nguyên n, chuyển đổi n về dạng thập phân
* Cho số n hãy tính biểu thức sau 1 + 2\*2\*2 + 3\*3\*3 + … + n^3
* Tính số thứ n trong dãy Fibonacci
* Cho số nguyên n, dòng đầu tiên in ra các chữ số của n từ trái sang phải, dòng thứ 2 in ra từ phải sang trái.
* Cho số nguyên dương n. Thực hiện các thao tác sau đây:

1. Lấy n chia cho 2 nếu n chia hết cho 2

2. Lấy n chia cho 3 nếu n chia hết cho 3

3. Giảm n đi 1 đơn vị.

4. Đếm số thao tác ít nhất để biến đổi n về 1.

* Kiểm tra mảng toàn chẵn: Cho mảng có n phần tử, kiểm tra xem mảng có tất cả phần tử là chẵn hay không
* Cho số nguyên n, kiểm tra xem n có tất cả chữ số là lẻ hay không?
* Cho số nguyên dương n, sử dụng đệ quy giải quyết các yêu cầu sau:

1. In ra dạng nhị phân của số n
2. In ra số đảo ngược