

BÀI TẬP ĐỆ QUY TRONG C++

1. Tìm số Fibonacci bằng đệ quy

Dãy fibonacci là dãy vô hạn các số tự nhiên bắt đầu bằng hai phần tử 0 và 1, các phần tử sau đó được thiết lập theo quy tắc mỗi phần tử luôn bằng tổng hai phần tử trước nó.

Hàm tìm số fibonacci bằng đệ quy

```
int fibonacci(int n)
{
    if (n < 0)
    {
        return -1;
    }
    else if (n == 0 || n == 1)
    {
        return n;
    }
    else
    {
        return fibonacci(n - 1) + fibonacci(n - 2);
    }
}
```

2. Tìm ước chung lớn nhất và bội chung nhỏ nhất bằng đệ quy

Ước chung lớn nhất và bội chung nhỏ nhất của 2 số là các khái niệm khá phổ biến trong toán học:

- **Ước chung lớn nhất của 2 số:** Là số lớn nhất mà 2 số đó cũng chia hết
- **Bội chung nhỏ nhất của 2 số:** Là số nhỏ nhất cùng chia hết cho 2 số đó

Hàm tìm ước chung lớn nhất và bội chung nhỏ nhất bằng đệ quy:

```
//Tìm ước chung lớn nhất

int UCLN(int a, int b) {

    if (b == 0)

        return a;

    return UCLN(b, a % b);

}

//Tìm bội chung nhỏ nhất

int BCNN(int a, int b) {

    return (a * b) / USCLN(a, b);

}
```

3. In đảo ngược một số nguyên dương n

Cho một số nguyên dương n, hãy viết hàm đệ quy để in ra màn hình đảo ngược của số nguyên dương đấy.

Hàm in đảo ngược một số nguyên dương n:

```
void InDaoNguoc(int n)

{

    if(n!=0)

    {

        cout<<n%10;

        InDaoNguoc(n/10);

    }

}
```

```
}  
  
}
```

4. In ra dạng nhị phân của số nguyên dương n

Cho một số nguyên dương n, hãy viết hàm in ra dạng nhị phân của số nguyên dương đó

Hàm in ra dạng nhị phân của số nguyên dương n

```
void NhiPhan(int n)  
{  
    if(n!=0)  
    {  
        NhiPhan (n/2);  
        cout<<n%2;  
    }  
}
```

5. Bài toán tính giai thừa

Cho **n** là một số tự nhiên ($n \geq 0$). Hãy tính giai thừa của n (**n!**) biết rằng $0! = 1$ và $n! = (n-1)! * n$.

Hướng dẫn:

Giả sử cần tính $5!$, lúc này quy luật của nó sẽ là.

- $5! = 4! * 5$
- $4! = 3! * 4$
- $3! = 2! * 3$
- $2! = 1! * 2$
- $1! = 1$

Thay các vế vào ta sẽ được quy luật: $5! = 1 * 2 * 3 * 4 * 5$.

Giả sử ta có hàm tính giai thừa của một số tên là GT, lúc này thay vào công thức trên ta được như sau:

Giả sử cần tính $5!$, lúc này quy luật của nó sẽ là.

- $5! = 4! * 5$
- $4! = 3! * 4$
- $3! = 2! * 3$
- $2! = 1! * 2$
- $1! = 1$

Thay các vế vào ta sẽ được quy luật: $5! = 1 * 2 * 3 * 4 * 5$.

Giả sử ta có hàm tính giai thừa của một số tên là GT, lúc này thay vào công thức trên ta được như sau:

Ví dụ

```
1#include<iostream>
2using namespace std;
3int GiaiThua(int n) {
4    // Trường hợp người dùng nhập
5    if (n == 1)
6        return 1;
7    else
8        return (n * GiaiThua(n - 1));
9}
10
11int main()
12{
13    int n;
14    while(true) {
15        cout << "Nhap so n: ";
16        cin >> n;
17        //Nhap n nho hon 0 de thoat khoi vong lap
18        if(n < 0) {
19            cout << " So am khong co giai thua" << endl;
20            break;
21        }
22        cout << " Giai thua cua " << n << " la: " << GiaiThua(n) << endl;
23    }
24    return 0;
25}
```

Và kết quả sau khi thực thi chương trình trên như sau:

```
Nhap so n: 4
Giai thua cua 4 la: 24
Nhap so n: 3
Giai thua cua 3 la: 6
Nhap so n: 2
Giai thua cua 2 la: 2
Nhap so n: 0
Giai thua cua 0 la: 1
Nhap so n: -5
So am khong co giai thua
```

FREEUTS

6. Dãy Fibonacci

Dãy Fibonacci là dãy vô hạn các số tự nhiên. Số Fibonacci thứ **n**, ký hiệu **F(n)**, được định nghĩa như sau :

- **F(n) = 0**, nếu **n = 0**
- **F(n) = 1**, nếu **n = 1**
- **F(n) = F(n-1) + F(n-2)**, nếu **n > 1**

Yêu cầu: tính số fibonacci thứ n với n cho trước.

```
#include <iostream>
using namespace std;

int fibonacci(int number)
{
    if (number == 0)
        return 0; // điều kiện dừng
    if (number == 1)
        return 1; // điều kiện dừng
    return fibonacci(number - 1) + fibonacci(number - 2);
}

int main()
{
    // in dãy 15 số fibonacci
    for (int count = 0; count < 15; ++count)
        cout << fibonacci(count) << " ";

    return 0;
}
```

1
2
3
4
5

```
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
```

Output: 0 1 1 2 3 5 8 13 21 34 55 89 144 233 377

7. Viết chương trình C++ sử dụng khái niệm hàm đệ qui để tính tích sau:

$S_n = 11 * 22 * 33 * 44 * \dots * nn$

Lời giải:

Dưới đây là chương trình C++ để giải bài tập trên

```
#include <cstdlib>
#include <iostream>
#include <cmath>
using namespace std;
double bieuthuc(int);

int main(int argc, char *argv[])
{

    int n;
    cout<<"Nhap n bang: ";
    cin>>n;
    cout<<"Ket qua: "<<bieuthuc(n)<<endl;

    return 0;
}

double bieuthuc(int n){
    if(n==1) return 1; //co so co so
    else return(pow((float)n,n)* bieuthuc(n-1)); //co so chung
```

```
}
```

Chạy chương trình C++ trên sẽ cho kết quả như hình sau:

```
Nhap n bang: 4
Ket qua: 27648

-----
Process exited after 3.509 seconds with return value 0
Press any key to continue . . .
```