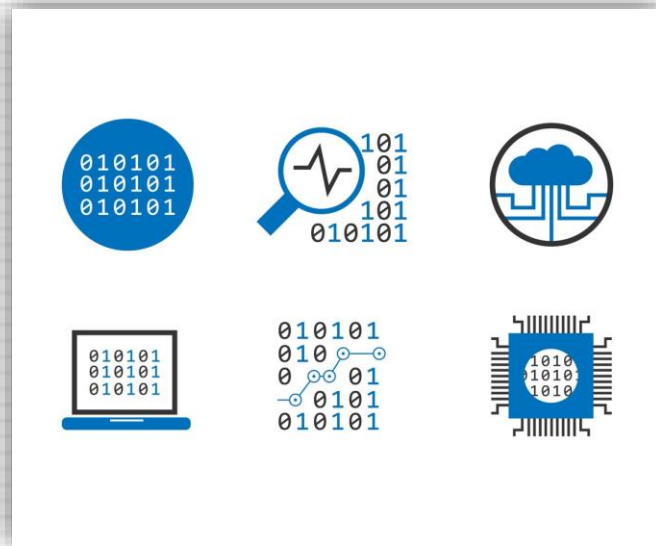


KỸ THUẬT LẬP TRÌNH

CHƯƠNG 3: ĐỆ QUY (RECURSION)

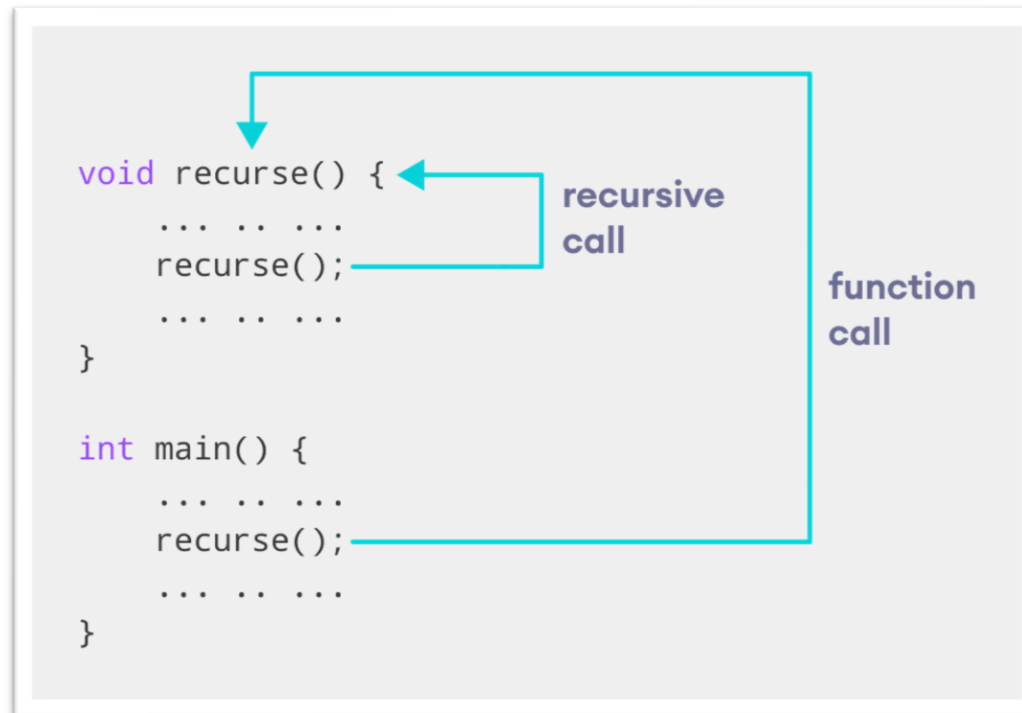


GV: Phạm Nguyễn Sơn Tùng

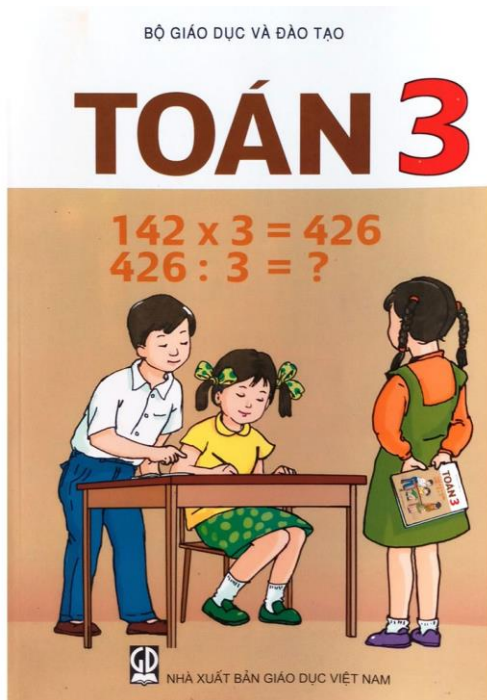
Email: pnstung@fit.hcmus.edu.vn

GIỚI THIỆU VỀ ĐỆ QUY

Định nghĩa: khi bạn viết một hàm, và bạn sử dụng lại hàm đó bên trong thân hàm, đó gọi là đệ quy.



GIỚI THIỆU VỀ ĐỆ QUY



GIỚI THIỆU VỀ ĐỆ QUY

Một hàm đệ quy đơn giản.

```
#include <iostream>
using namespace std;
void Ham(int n)
{
    if (n > 0)
    {
        cout << n << " ";
        Ham(n - 1);
    }
}
int main()
{
    int n = 5;
    Ham(n);
    return 0;
}
```

GIỚI THIỆU VỀ ĐỆ QUY

Một hàm đệ quy khác. Theo bạn kết quả sẽ như thế nào?

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

void Ham(int n)
{
    if (n > 0)
    {
        Ham(n - 1);
        cout << n << " ";
    }
}

int main()
{
    int n = 5;
    Ham(n);
    return 0;
}
```

CÁC LOẠI ĐỆ QUY

1. Đệ quy tuyến tính / đệ quy trực tiếp: là loại đệ quy có duy nhất 1 lời mời gọi hàm.

```
int Tong(int n)
{
    if (n == 1)
        return 1;
    return Tong(n - 1) + n;
}

int main()
{
    int n = 5;
    int ketqua = Tong(n);
    cout << ketqua;
    return 0;
}
```

CÁC LOẠI ĐỆ QUY

2. Đệ quy nhị phân: là loại đệ quy có 2 lời gọi làm lại chính nó.

```
long Fibo(int n)
{
    if (n == 0 || n == 1)
        return 1;
    return Fibo(n-1) + Fibo(n-2);
}
int main()
{
    cout << Fibo(5);
    return 0;
}
0 1 2 3 4 5
1 1 2 3 5 8
```

CÁC LOẠI ĐỆ QUY

3. Đệ quy hỗ tương: là loại đệ quy thân hàm này gọi tới làm kia và thân làm kia gọi tới hàm này.

```
void HamA(int n)
{
    if (n > 0)
    {
        cout << n << " ";
        HamB(n - 1);
    }
}
void HamB(int n)
{
    if (n > 1)
    {
        cout << n << " ";
        HamA(n / 2);
    }
}
```


CÁC LOẠI ĐỆ QUY

3. Đệ quy hỗ trợ: là loại đệ quy thân hàm này gọi tới làm kia và thân làm kia gọi tới hàm này.

```
int main()
{
    cout << HamA(6);
    return 0;
}
```

CÁC LOẠI ĐỆ QUY

4. Đệ quy phi tuyến: là loại đệ quy lời gọi hàm nằm bên trong vòng lặp.

Tính tổng $S(n) = S(1) + S(2) + S(3) + \dots + S(n-1)$

(Với $S(1) = 1$)

Ví dụ: Tính $S(5)$

$$S(5) = S(1) + S(2) + S(3) + S(4)$$

$$S(4) = S(1) + S(2) + S(3)$$

$$S(3) = S(1) + S(2)$$

$$S(2) = S(1) = 1$$

$$\rightarrow S(2) = 1$$

$$\rightarrow S(3) = 2$$

$$\rightarrow S(4) = 4$$

$$\rightarrow S(5) = 8$$

CÁC LOẠI ĐỆ QUY

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

int Tong(int n)
{
    if (n == 1)
        return 1;
    long s = 0;
    for (int i = 1; i < n; i++)
        s += Tong(i);
    return s;
}

int main()
{
    int ketqua = Tong(5);
    cout << ketqua;
}
```

CÁC LOẠI ĐỆ QUY ĐẶC BIỆT

5. Đệ quy đuôi: Là đệ quy có lời gọi nằm ở cuối cùng.

```
void Ham(int n)
{
    if (n == 0)
        return;
    else
        cout << n << endl;
    return Ham(n - 1);
}

int main()
{
    Ham(5);
    return 0;
}
```

CÁC LOẠI ĐỆ QUY ĐẶC BIỆT

5. Đệ quy đuôi: khác nhau với đệ quy bình thường như thế nào?

```
void Ham(int n)
{
    if (n == 0)
        return;
    Ham(n - 1);
    cout << n << endl;
}

int main()
{
    Ham(5);
    return 0;
}
```

CÁC LOẠI ĐỆ QUY ĐẶC BIỆT

6. Đệ quy lồng nhau: ít khi sử dụng vì thời gian chạy rất lâu.

```
int fun(int n)
{
    if (n > 100)
        return n - 10;
    return fun(fun(n + 11));
}

void main()
{
    cout << fun(95);
}
```

CÁC LOẠI ĐỆ QUY ĐẶC BIỆT

6. Đệ quy lồng nhau: Hàm Ackerman trong toán logic.

$$A(m, n) = \begin{cases} n + 1 & \text{if } m = 0 \\ A(m - 1, 1) & \text{if } m > 0 \text{ and } n = 0 \\ A(m - 1, A(m, n - 1)) & \text{if } m > 0 \text{ and } n > 0 \end{cases}$$

```
int ack(int m, int n)
{
    if (m == 0)
        return n + 1;
    else if ((m > 0) && (n == 0))
        return ack(m - 1, 1);
    else if ((m > 0) && (n > 0))
        return ack(m - 1, ack(m, n - 1));
}
```

CÁC LOẠI ĐỆ QUY ĐẶC BIỆT

Đệ quy còn được áp dụng vào rất nhiều phương pháp lập trình khác:

- **Đệ quy trên danh sách.**
- Backtracking (quay lui)
- Divide & Conquer (chia để trị)
- Dynamic Programming (Quy hoạch động)

CÁC BÀI TOÁN ĐỆ QUY KHÁC

Bài toán: Tổ hợp là chọn những phần tử từ một nhóm lớn mà không phân biệt thứ tự.

$$C(n, k) = \frac{n!}{(n - k)!k!}$$

Ví dụ 1: xác suất lấy 5 lá bài trong bộ bài 52 lá.

Ví dụ 2:

$$C_k^n = \frac{n!}{k! * (n - k)!}$$

$$C_6^{45} = \frac{45!}{6! * (45! - 6!)} = 8145060$$

CÁC BÀI TOÁN ĐỆ QUY KHÁC

Cách 1: Giải bài tổ hợp chập k của n phần tử. Sử dụng 2 hàm khác nhau.

```
int GiaiThua(int n)
{
    if (n == 1)
        return 1;
    return n * GiaiThua(n - 1);
}
int C(int n, int k)
{
    int a = GiaiThua(n);
    int b = GiaiThua(n - k);
    int c = GiaiThua(k);
    return a/(b*c);
}
```

CÁC BÀI TOÁN ĐỆ QUY KHÁC

Cách 2: Giải với công thức quy nạp.

```
int C(int n, int k)
{
    if (k == 0 || k == n)
        return 1;
    if (k == 1)
        return n;
    return C(n - 1, k - 1) + C(n-1, k);
}
```