

THỰC HÀNH CƠ SỞ LẬP TRÌNH

FUNCTION 2

[HTTPS://WWW.TUTORIALSPOINT.COM/CPLUSPLUS/_CPP_FUNCTIONS.HTM](https://www.tutorialspoint.com/cplusplus/cpp_functions.htm)
[HTTPS://WWW.TUTORIALSPOINT.COM/CPROGRAMMING/C_RECURSION.HTM](https://www.tutorialspoint.com/cprogramming/c_recursion.htm)

FOR YOUR INFORMATION

- Nguyễn Mạnh Duy
 - Email: nmduy881994@gmail.com
- Thiều Bảo Trân
 - Email: thieubaotran@gmail.com
- Title: [CSLT]

CONTENTS

- Parameters - Tham số
 - Call by value – Truyền tham trị
 - Call by reference – Truyền tham chiếu
- Recursion – đệ quy
- Exercises – Thực hành (Xem file PDF)
- STEP@TUAT Introduction

PARAMETERS

- **Call by Value – Truyền Tham Trị**
 - **Copy giá trị** của các tham số vào hàm để tính toán
 - Sau khi kết thúc hàm thì giá trị các tham số **không bị ảnh hưởng**
- **Call by Reference – Truyền Tham Chiếu**
 - **Copy địa chỉ** của các tham số vào hàm để tính toán
 - Sau khi kết thúc hàm thì giá trị các tham số **có thể bị ảnh hưởng**
 - Trong phần khai báo và định nghĩa hàm, trước tên biến có **ký tự ‘ & ’**
 - Khi gọi hàm thì không cần ký tự ‘ & ’

VALUE – REFERENCE EXAMPLE

Tham Trị

```
void Swap (int a, int b){  
    int temp = a;  
    a = b;  
    b = temp;  
}
```

Tham Chiếu

```
void Swap (int &a, int &b){  
    int temp = a;  
    a = b;  
    b = temp;  
}
```

a = 2
b = 3

```
void main(){  
    int a = 2;  
    int b = 3;  
    Swap(a,b);  
    printf("%d %d\n",a,b);  
}
```

a = 3
b = 2

PARAMETERS

- **Mảng** hay **con trỏ** truyền vào hàm đều **có thể bị thay đổi**

Hàm Nhập mảng

```
void InputArray(int a[], int &na){
    printf("Nhap Mang\nNhap so luong phan tu: ");
    scanf("%d", &na);
    printf("Nhap gia tri cac phan tu: \n");
    for(int k = 0; k < na; k++){
        printf("Phan tu [%d]: ",k);
        scanf("%d", &a[k]);
    }
}
```

Tại sao là **int &na**?

Hàm Xuất mảng

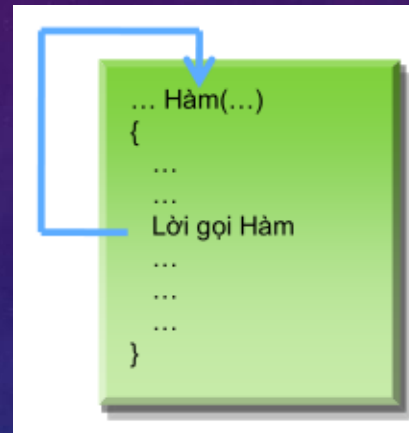
```
void PrintArray(int a[], int na){
    printf("Xuat Mang\nMang co %d phan tu\n",na);
    printf("Gia tri cac phan tu trong mang:\n");
    for (int k = 0; k < na; k++){
        printf("Phan tu [%d]: %d\n",k,a[k]);
    }
}
```

Tại sao chỉ là **int na**?

Hàm Main

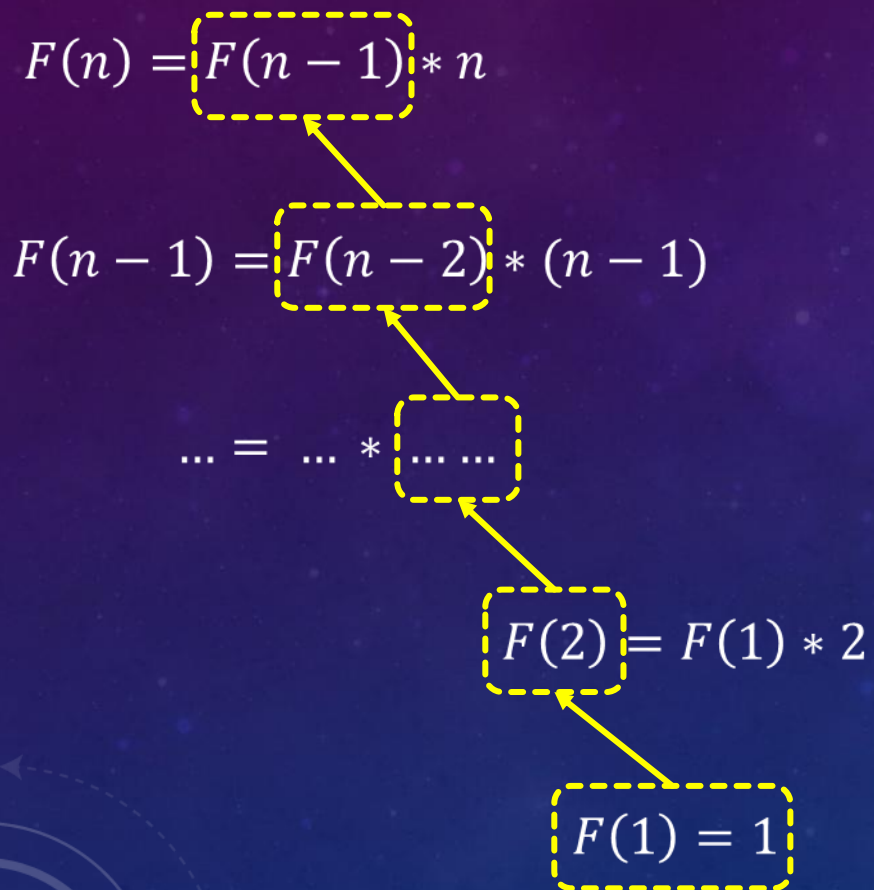
```
void main(){
    int a[20] = {};
    int na = 0;
    printf("----- NHAP MANG ----- \n");
    InputArray(a,na);
    printf("----- XUAT MANG ----- \n");
    PrintArray(a,na);
}
```

RECURSION – ĐỆ QUY



- Hàm mà gọi lại chính nó
- Phải thỏa 2 điều kiện
 - Tồn tại bước đệ quy (có gọi lại chính hàm đó)
 - Tồn tại điều kiện dừng (tránh loop vô hạn)

RECURSION – ĐỆ QUY



Ví dụ: Tính $F(n) = n! = 1 * 2 * \dots * n$

$$F(n) = n * (n-1) * \dots * 2 * 1 = n * (n-1)!$$

$$F(n) = n * F(n-1) \quad \text{Bước đệ quy}$$

$$F(1) = 1 \quad \text{Điều kiện dừng}$$

```
int FacN(int n){  
    if (n <= 1)  
        return 1;  
    else  
        return n * FacN(n-1);  
}
```


EXERCISES – THỰC HÀNH (XEM FILE PDF)

STEP@TUAT INTRODUCTION

- Học bổng trao đổi sinh viên với Nhật Bản
- Thời gian trao đổi: 1 năm
- 80000 JPY / tháng (~ 800 USD)
- Văn hóa, xã hội, con người Nhật Bản
- Nghiên cứu đề tài với giáo sư tự chọn
- Có thể chuyển được tín chỉ về VN
- Yêu cầu:
 - Điểm TB ~ 8.0
 - Anh Văn ~ TOEIC 550



EXERCISES – GỢI Ý

```
void Swap(int &a, int &b);
void InputArray(int a[], int &na);
void PrintArray(int a[], int na);
int MinArray(int a[], int na);
int MaxArray(int a[], int na);
int SumArray(int a[], int na);
float MeanArray(int a[], int na);
void SortArray(int a[], int na);
void InsertElementArray(int a[], int &na, int values, int position);
void RemoveElementArray(int a[], int &na, int position);
void bai1();
int SumN(int n);
int FacN(int n);
int Fibonacci(int n);
void bai2();
void PascalTriangle(int m);
void bai3();
void CaesarCipher(char a[], int na, int k);
void bai4();
```

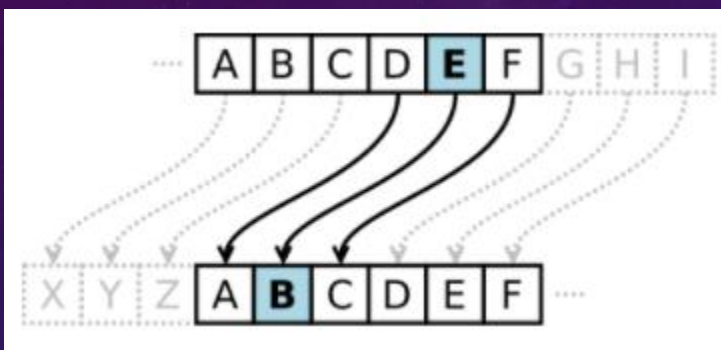
```
void main(){
    printf("===== BAI 1 =====\n");
    bai1();
    printf("===== BAI 2 =====\n");
    bai2();
    printf("===== BAI 3 =====\n");
    bai3();
    printf("===== BAI 4 =====\n");
    bai4();
}
```


EXERCISES – PASCAL TRIANGLE

	$j=0$	$j=2$	$j=4$	
$i=0$	1	0	0	0
	1	1	0	0
$i=2$	1	2	1	0
	1	3	3	1
$i=4$	1	4	6	4
	1	5	10	10

- Nếu $n = 0$, in ra màn hình “ 1 “
- Với $n \geq 1$
 - Tạo mảng 2D kích thước $(n+1) \times (n+1)$
 - $A[0][0] = 1$
 - Đi từng hàng ($i \leq n$) và thay đổi
 - $A[i][0] = A[i][i] = 1$
 - Tính $A[i][j]$ ($0 < j < i$) bằng cách
 - $A[i][j] = A[i-1][j] + A[i-1][j-1]$

EXERCISES – CAESAR CIPHER



Shift với hệ số 23

- Dùng mã ASCII để làm
- Với mỗi ký tự $B[i] = B[i] + \text{hệ số}$
- Lưu ý trường hợp vượt quá giá trị cho phép
- $B[i] \in [65, 90]$ (Mã ASCII của các ký tự in hoa từ A – Z)
- <https://learncryptography.com/classical-encryption/caesar-cipher>

HOW TO SUBMIT

- **Deadline:** 23h59 23/06/2017
- **Link nộp bài:** <https://goo.gl/IMwaQY>
- Chỉ nộp file .cpp và .h (nếu có). Gộp tất cả file này vào một folder. Đặt tên folder là MSSV. Sau đó nén lại thành file .rar hoặc .zip. (Tên file sau khi nén vẫn là MSSV)
- Truy nhập **Link nộp bài** (bên trên), sau đó tải file đã nén lên web.

HOW TO SUBMIT

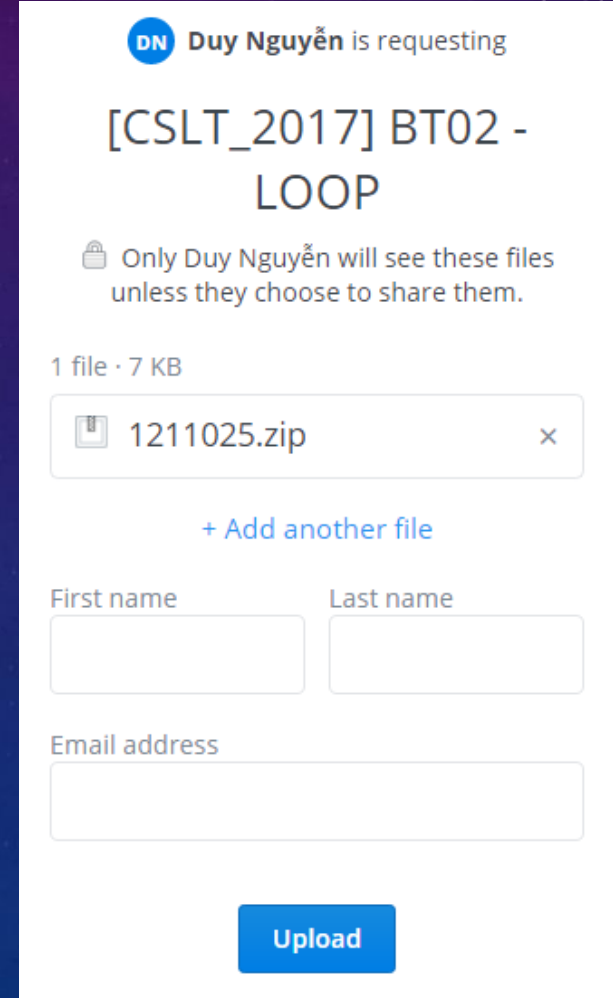
First name: Tên người làm bài

Last name: Họ người làm bài

Email: Email người làm bài

Nếu muốn nộp lại bản mới thì làm tương tự.


Không đổi tên file, folder, không đổi thông tin đã điền như lần đầu (First name, Last name, Email)





The screenshot shows a web interface for file submission. At the top, it says 'DN Duy Nguyễn is requesting'. Below this is the title '[CSLT_2017] BT02 - LOOP'. A lock icon indicates that only the user 'Duy Nguyễn' will see the files unless they choose to share them. It shows '1 file · 7 KB' and a file named '1211025.zip' with a close button. There is a link '+ Add another file'. Below are input fields for 'First name', 'Last name', and 'Email address'. At the bottom is a blue 'Upload' button.

DN Duy Nguyễn is requesting

[CSLT_2017] BT02 -
LOOP

 Only Duy Nguyễn will see these files
unless they choose to share them.

1 file · 7 KB

 1211025.zip 

[+ Add another file](#)

First name

Last name

Email address

[Upload](#)



THE END