File văn bản và file nhị phân

* File văn bản: lưu dưới dạng các ký tự ASCII – ký tự đọc được
  + Đọc bằng 1 công cụ soạn thảo văn bản bất kỳ
  + Tốn bộ nhớ, kém bảo mật
* File nhị phân: Nội dung lưu bằng chuỗi nhị phân có định dạng trước
  + Bảo mật, muốn đọc thì phải biết được định sạng của file (thường chỉ có chương trình phù hợp mới đọc được)
  + Tiết kiệm bộ nhớ hơn

Chọn chế độ xử lý file phù hợp: r, w, a. r+, w+, a+

Mở file cần truyền vào - đường dẫn + chế độ

Khi mở hệ điều hành sẽ nạp block đầu tiên của file từ bộ nhớ ngoài vào trong RAM, trả về địa chỉ ô nhớ đầu tiên của block đó trong RAM (kích thước 1 block = kích thước 1 sector khi format ổ cứng)

Sau khi sử dụng phải đóng file

* Báo hiệu cho các ứng dụng khác là file đã dùng xong
* Ghi những sửa đổi nếu có của file vào ổ cứng (nếu không đóng file, các sửa đổi có thể chỉ diễn ra trên RAM)

Để báo hiệu kết thúc file, hằng số EOF sẽ được tự động thêm vào cuối file bởi HDH

Khi đọc file, cần check EOF để xem đã đọc tới cuối file hay chưa

**Bài tập 1**. Viết chương trình đọc 1 file văn bản với tham số tên file được đọc từ dòng lệnh. In ra màn hình số lượng **ký tự viết Hoa**, **thường** và **các dấu câu (,.?)** có trong đoạn văn đó.

Thao tác với file văn bản

* Thao tác theo từng ký tự
* Thao tác theo từng dòng - line
* Thao tác theo khối

**VD 1**. Viết chương trình đọc nội dung file văn bản và in ra số lượng dòng (KHÔNG phải số lượng câu) của 1 file văn bản với tham số truyền vào từ dòng lệnh.

Chú ý: Đọc file theo dòng (buffer)

* Tham số MAX\_LEN sẽ ảnh ưởng tới số lượng dòng đếm được  
  Nếu set MAX\_LEN nhỏ quá --> số lượng dòng lớn  
  Nếu set MAX\_LEN đủ lớn --> số lượng dòng thực sự

Ví dụ 2. Cho 1 file văn bản chỉ chưa các số nguyên. Hãy viết chương trình để đếm số lượng số, tổng các số và giá trị trung bình cộng. In ra màn hình kết quả tính được

Các số được ngăn cách với nhau bởi ít nhất 1 dấu cách trống, enter hoặc tab

12 23 –45

23 12 –4 2 45

3

**Bài tập 2**. Viết chương trình đếm số lượng câu trong văn bản (khác với số lượng dòng). Các câu sẽ được kết thúc bởi dấu . Hoặc ! Hoặc ?

In ra màn hình số lượng câu đếm được.

Tên file được truyền trong tham số dòng lệnh

Chú ý: Cần phân biệt với dấu … liền nhau (độ dài câu phải >1)

VD. Cho đoạn văn sau

IBM is the global leader in business transformation through an open hybrid cloud platform and AI, serving clients in more than 170 countries around the world. Today 47 of the Fortune 50 Companies rely on the IBM Cloud to run their business, and IBM Watson enterprise AI is hard at work in more than 30,000 engagements. IBM is also one of the world’s most vital corporate research organizations, with 28 consecutive years of patent leadership. Above all, guided by principles for trust and transparency and support for a more inclusive society, IBM is committed to being a responsible technology innovator and a force for good in the world...

Đoạn văn có 4 câu

===================================

BÀI TẬP VỀ NHÀ

**Bài 1**. Chương trình mã hóa dịch offset ký tự - mã Caesar

Với các ký tự là chữ Hoa hoặc thường ta sẽ dịch vòng trong với độ lệch là offset.  
VD. Offset là 2  
a-->c, b-->d,…, y-->a, z->b  
A-->C, B-->D,…, Y-->A, Z->B

Các ký tự khác vẫn giữ nguyên

Xây dựng chương trình mã hóa 1 file văn bản và giải mã

**Bài tập 2.**

Danh sách sinh viên được cho bởi file danhsach.txt gồm có

Dòng đầu là : số lượng sinh viên

Các dòng tiếp theo là mã sinh viên và họ tên

===================

3

12345 Nguyen Van A

23224 Tran Thi B

23432 Le Van C

==================

Thông tin những sinh viên đi học được lưu trong file log gồm có mã sinh viên (kiểu số nguyên).

================

12345

23432

=================

Hãy viết chương trình đọc vào file văn bản đó và in ra những sinh viên nào Vắng mặt

VD. Với file mẫu ở trên thì người vắng mặt sẽ là 23432 Le Van C

**Bài tập 3**. Viết chương trình tìm và in ra tần số xuất hiện lớn nhất của số nguyên cho bởi file digit.txt

Nội dung file là

12 34 2 4 52 45 4 2 45 33 12 34 21 5 21 13

Mỗi số được ngăn cách bởi ít nhất 1 dấu cách trống

+ Dùng 1 mảng 2 chiều hoặc 2 mảng 1 chiều

2 mảng 1 chiều

int values[MAX\_SIZE]={0};

int freqs [MAX\_SIZE]={0};

Int cur\_size = 0; // so luong so hien co

Số thứ i sẽ có giá trị là values[i] và tần số là freqs [i]

fscanf(fptr,"%d",&digit)

int check =0;

for(int j=0; j< cur\_size; j++)

if(values[j]== digit) {

freqs[j]++;

check =1;

}

if(check==0)//so do ko nam trong danh sach so da co

{

values[cur\_size] = digit;

freqs[cur\_size]=1;

cur\_size++;

}