

# **CHƯƠNG 4. THAO TÁC VỚI TỆP TIN**

**Giáo viên : ThS. Lê Đức Quang**

**Chuyên môn : KTHT và MTT**

**Đơn vị : Bộ môn KTHT & MTT**

# Nội dung

---

- 5.1 Giới thiệu chung về tệp tin
- 5.2 Giới thiệu các hàm chuẩn thao tác với tệp tin
- 5.3 Thao tác mở/đóng tệp, kiểm tra lỗi
- 5.4 Đọc/ghi tệp

## 5.1 Giới thiệu

---

- Tập tin: đơn giản là **một dãy các byte dữ liệu** được ghi **trên ổ đĩa**
  - Mỗi byte gồm 8 bit, có giá trị từ 0 – 255
  - Số byte của dãy chính là **độ dài của tệp**
- Tác dụng: lưu trữ dữ liệu
  - Lưu trữ dữ liệu của chương trình: input và output
  - Lưu trữ các tài liệu, văn bản, các dữ liệu âm thanh, hình ảnh...
- Tại sao cần thao tác với tệp trong chương trình?
  - Vì chương trình cần dữ liệu mà tệp là nơi lưu trữ dữ liệu

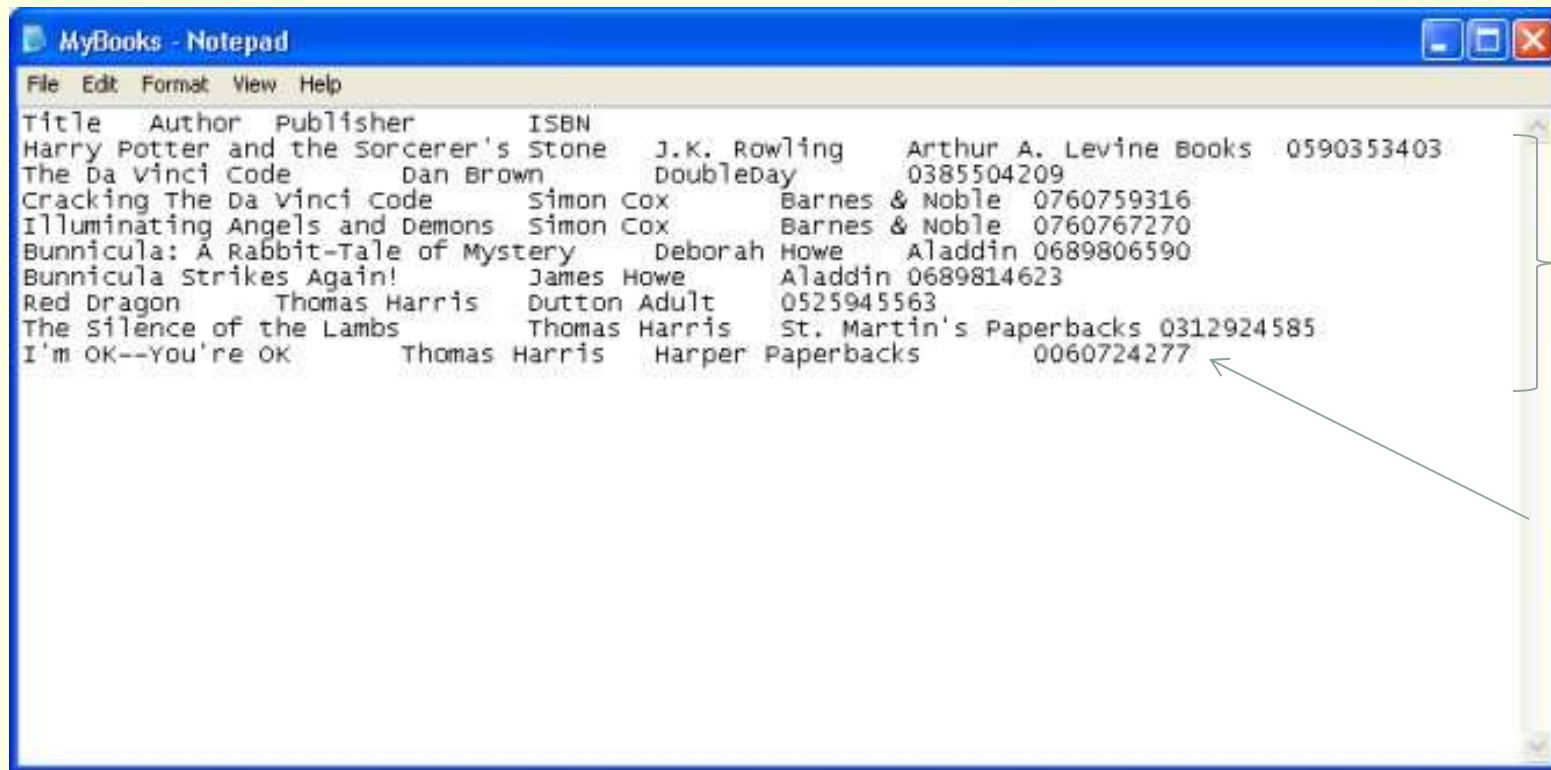
# Tiếp

---

- Các thao tác trên tệp (chung , không phụ thuộc ngôn ngữ lập trình)
  - Tạo một tệp tin mới
  - Mở một tệp tin đã có trên ổ đĩa
  - Đọc dữ liệu từ tệp lưu vào bộ nhớ của chương trình
  - Ghi dữ liệu từ bộ nhớ chương trình vào tệp
  - Đóng tệp để kết thúc
- Kiểu tệp
  - Quy định bởi phần tên mở rộng và định dạng dữ liệu bên trong tệp
  - Nói chung có thể chia thành **tệp dữ liệu văn bản** và **tệp dữ liệu nhị phân**

# Tệp văn bản

Mở tệp văn bản bằng chương trình Notepad



Các  
dòng  
văn  
bản

Kết  
thúc  
tệp

- Tệp văn bản có thể chia thành các dòng, mỗi dòng gồm nhiều kí tự
- Kết thúc mỗi dòng là cặp kí tự CR (13) và LF(10)

# Tập nhị phân

Mở tệp nhị phân bằng các chương trình như HexEditor, BinView

- Tập gồm dãy các byte
- Hình bên thể hiện các byte dưới dạng số Hexa (hệ 16)
- Để tiện theo dõi nên chương trình hiển thị mỗi dòng 16 byte

[illegible]

## 5.2 Giới thiệu các hàm chuẩn thao tác với tệp tin

- Thư viện chuẩn **fstream** cung cấp 3 kiểu dữ liệu mới:

Kiểu dữ liệu	Miêu tả
ofstream	Kiểu dữ liệu dùng để biểu diễn Output File Stream và được sử dụng để tạo các file và để ghi thông tin tới các file đó.
ifstream	Kiểu dữ liệu dùng để biểu diễn Input File Stream và được sử dụng để đọc thông tin từ các file
fstream	Kiểu dữ liệu dùng để biểu diễn File Stream và có các khả năng của cả ofstream và ifstream, nghĩa là nó có thể tạo file, ghi thông tin tới file và đọc thông tin từ file

# Hàm dành cho các thao tác chung

---

- Các hàm chuẩn thao tác với tệp tin được gọi thông qua các đối tượng `ifstream`, `ofstream`, `fstream` bao gồm:
  - `open()` dùng để mở tệp / tạo tệp mới
  - `close()` dùng để đóng tệp
  - `fail()` dùng để kiểm tra xem đối tượng `fileInput` đã liên kết được đến file cần mở hay chưa.
  - `eof` để kiểm tra đã đến cuối tệp hay chưa
  - `getline()` dùng để đọc cả dòng trong tệp



## 5.3 Thao tác mở/đóng tệp

### a) Thao tác mở tệp.

Để mở tệp trước tiên chúng ta phải khai báo biến thuộc đối tượng ifstream, ofstream, fstream.

VD: `ifstream ten_bien; //khai báo biến tệp để đọc`  
`ofstream ten_bien; //Khai báo biến tệp để ghi`

**Mở tệp:** sử dụng hàm `open()` có kiểu như sau:

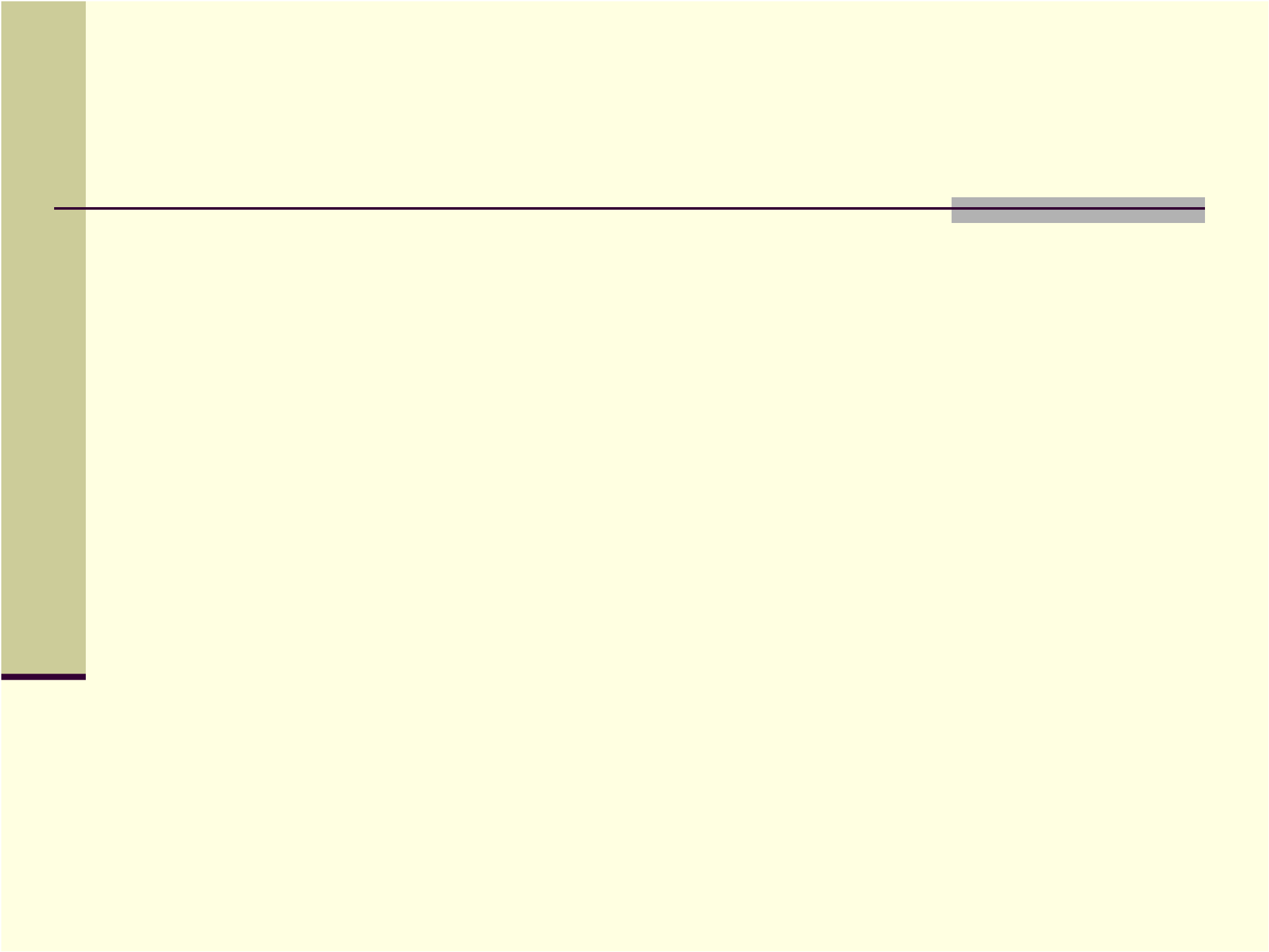
```
ten_bien.open(char* <tên tệp>, ios::che_do);
```

- Tham số thứ nhất : tên tệp (hoặc đường dẫn tệp)
- Tham số thứ hai: kiểu xuất / nhập

# Các chế độ mở tệp

Chế độ	Miêu tả
ios::app	Mở 1 tệp để ghi bổ sung theo kiểu văn bản. Nếu tệp chưa tồn tại thì tạo tệp mới.
ios::in	Mở 1 tệp để đọc theo kiểu văn bản. Tệp cần tồn tại nếu không sẽ có lỗi.
ios::out	Mở 1 tệp để ghi theo kiểu văn bản. Nếu tệp đã tồn tại, nó sẽ bị xóa.

**Chú ý:** có thể sử dụng toán tử “|” để kết hợp các chế độ



## 5.3 Thao tác mở/đóng tệp

---

### b) Thao tác đóng tệp.

Để đóng tệp sử dụng hàm `close()`

Cú pháp: `ten_bien.close();`

#### •Tác dụng của hàm

- Đẩy dữ liệu trong vùng đệm của tệp lên đĩa khi ở chế độ ghi
- Xóa dữ liệu trong vùng đệm khi ở chế độ đọc
- Giải phóng biến tệp

# Hàm kiểm tra lỗi

---

## c) Sử dụng hàm `fail()` để kiểm tra lỗi mở tệp.

•Cú pháp: `ten_bien.fail()`

- Hàm trả về **true** nếu đối tượng `ifstream` liên kết đến file không thành công.
- Hàm trả về **false** nếu đối tượng của `ifstream` liên kết đến file thành công

# Hàm kiểm tra kết thúc tệp

---

## d) Để kiểm tra kết thúc tệp khi thao tác, sử dụng hàm **eof()**

Cú pháp: `ten_bien.eof()` ;

- Tác dụng của hàm để kiểm tra kết thúc tệp tin
- Nếu đã gặp cuối tệp khi đọc, hàm trả về giá trị khác 1
- Ngược lại, khi chưa đọc đến cuối tệp, hàm trả về giá trị 0

## 5.4 Đọc/ghi tệp tin kiểu văn bản

---

### a) Ghi dữ liệu

Để ghi thông tin tới một file: sử dụng toán tử: “<<”, cách sử dụng giống như khi chúng ta sử dụng toán tử đó để hiển thị thông tin ra màn hình. Chỉ có một điểm khác nhau là chúng ta sử dụng một đối tượng **ofstream** hoặc **fstream** trong C++ thay cho đối tượng **cout** trong C++.

# Ví dụ về ghi dữ liệu từ tệp

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

int main()
{
    ofstream f; //Khai báo đối tượng để ghi từ tệp
    int i;
    f.open("text.txt",ios::out); //Mở tệp để ghi
    if(f.fail())
    {
        cout<<"Loi khi tao tep";
        return 1; // Kết thúc hàm main khi gặp lỗi
    }
    f<<"Cac dong"<<endl; //Ghi dữ liệu vào tệp
    for(i=0;i<2;i++) //
        f<<"Dong " <<i<<endl; //Ghi dữ liệu vào tệp
    f.close(); //Đóng tệp
    return 0;
}
```



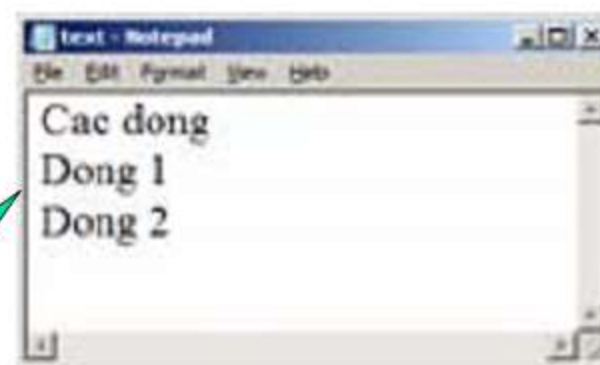
# Ví dụ về ghi dữ liệu từ tệp

Chương trình trên sẽ tạo ra tệp văn bản tên là **text** gồm 3 dòng với nội dung như sau:

Cac dong

Dong 1

Dong 2



Dùng trình notepad  
mở file **text** ra xem  
nội dung của nó

## 5.4 Đọc/ghi tệp tin kiểu văn bản

---

### a) Đọc dữ liệu

Để đọc thông tin từ một file: sử dụng toán tử: “>>”, cách sử dụng giống như khi chúng ta sử dụng toán tử đó để nhập thông tin từ bàn phím. Chỉ có một điểm khác nhau là chúng ta sử dụng một đối tượng **ifstream** hoặc **fstream** trong C++ thay cho đối tượng **cin** trong C++.

# Ví dụ về đọc dữ liệu từ tệp

Giả sử có tệp văn bản "da\_giac.sl" chứa thông tin về một đa giác. Tệp gồm  $n+1$  dòng với nội dung như sau:

Dòng 1:  $n$  (số đỉnh)

Dòng 2:  $x_1 y_1$  (tọa độ đỉnh 1)

Dòng 3:  $x_2 y_2$  (tọa độ đỉnh 2)

....

Dòng  $n+1$ :  $x_n y_n$  (tọa độ đỉnh  $n$ )

Chương trình sau sẽ đọc số đỉnh và tọa độ các đỉnh từ tệp "da\_giac.sl"



# Ví dụ về đọc dữ liệu từ tệp

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

int main()
{
    ifstream f; //Khai báo đối tượng để đọc từ tệp
    int i,n;
    int x[20],y[20];
    f.open("da_giac.sl",ios::in); //Mở tệp để đọc
    if(f.fail())
    {
        cout<<"Loi khi mo tep";
        return 1; // ket thuc ham main khi gap loi
    }
    f>>n; //Đọc từ tệp
    for(i=0;i<n;i++) //
        f>>x[i]>>y[i]; //Đọc từ tệp
    f.close();
    cout<<"To do cac dinh cua da giam la:\n";
    for(i=0;i<n;i++)
        cout<<x[i]<<" "<<y[i]<<endl;
    return 0;
}
```

# Trình tự bước thao tác với tệp

---

## 1. Khai báo đối tượng để đọc hoặc ghi từ tệp

```
ifstream f;  
ofstream f;
```

## 2. Mở tệp (đọc / ghi / cả đọc cả ghi)

```
f.open("Input.dat", ios::out|ios::in);  
if(f.fail())  
    cout<<"Loi khi mo tep";
```

...

## 3. Đọc / ghi dữ liệu

Dùng toán tử ">>" để đọc dữ liệu và toán tử "<<" để ghi dữ liệu vào tệp.

- Chú ý kiểm tra kết thúc tệp nếu cần thiết (chủ yếu là khi đọc tệp)

## 4. Đóng tệp

```
f.close();
```

# Bài tập

---

## Bài 1: Viết chương trình

- Nhập từ bàn phím N số thực lưu vào một mảng 1 chiều ( $4 < N < 50$  và N nhập từ bàn phím)
- Ghi mảng số thực ra một tệp văn bản “daysothuc.txt” theo quy cách
  - Dòng đầu ghi giá trị N
  - Các dòng tiếp theo ghi các số trong dãy, mỗi số cần 5 vị trí và có 2 chữ số thập phân, 8 số trên 1 dòng

# Bài tập

---

## Bài 2: Viết chương trình

- Mở tệp “daysothuc.txt” để đọc dãy số thực từ tệp, và lưu vào mảng số thực
- Sắp xếp mảng số thực tăng dần
- Lưu dãy số đã sắp xếp ra một tệp mới “daysothuc\_tang.txt”

# Bài tập

---

Bài 3: Nội dung giống bài 1 nhưng là nhập ma trận  $N \times M$ , phần tử số nguyên từ bàn phím và ghi ra tệp “matran.txt” theo quy cách

- Dòng 1:  $N$   $M$  (là số hàng, số cột của ma trận)
- Từ các dòng tiếp theo, mỗi dòng ghi 1 hàng của ma trận, mỗi phần tử cần 4 vị trí



# Bài tập

---

## Bài 4: Viết chương trình

- Mở tệp “matran.txt” để đọc ma trận số nguyên từ tệp, và lưu vào mảng số thực (tự tìm số hàng và số cột)
- Hiển thị mảng số thực ra màn hình.
- Xây dựng véc tơ  $s$  là tổng từng cột của ma trận vừa đọc.
- Lưu véc tơ  $s$  vào tệp mới “tong\_cot.txt”