

**ĐỀ KIỂM TRA THỰC HÀNH**  
**HỌC PHẦN: KỸ THUẬT LẬP TRÌNH**

**Bài 1. (2 điểm)**

Dãy số  $x_n$  được định nghĩa như sau:

$$x_0 = 1; x_1 = 1;$$

$$x_n = nx_0 + (n-1)x_1 + (n-2)x_2 + (n-3)x_3 + \dots + x_{n-1}, \forall n > 1.$$

Viết chương trình tính  $x_n$  bằng phương pháp đệ quy.

**Bài 2. (4 điểm)**

Kết quả của một xét nghiệm vi rút covid-19 có hai trường hợp: 0: âm tính và 1: dương tính. Để kiểm tra độ chính xác của một công cụ xét nghiệm mới phát minh, người ta thực hiện như sau: lấy các mẫu đã biết kết quả xét nghiệm và ghi nhãn 0 hoặc 1 trên các mẫu này (tương ứng âm tính hoặc dương tính), các nhãn này gọi là true label, sau đó họ dùng bộ công cụ mới kiểm tra lại trên các mẫu và ghi nhận lại kết quả, các nhãn kết quả được gọi là predicted label. Cuối cùng, họ so sánh kết quả tương ứng giữa true label và predicted label, thực hiện thống kê việc trùng khớp kết quả là bao nhiêu và đưa ra kết luận về độ chính xác của công cụ xét nghiệm mới. Kết quả so khớp sẽ phân thành 4 loại được ghi trên 1 ma trận gọi là Confusion Matrix:

Confusion Matrix		Predicted label	
		1: dương tính	0: âm tính
True label	1: dương tính	TP (dương tính thật)	FN (âm tính giả)
	0: âm tính	FP (dương tính giả)	TN (âm tính thật)

- TP (dương tính thật): Khi kết quả xét nghiệm (predicted label) và kết quả gắn nhãn (true label) trùng khớp nhau đều là 1 (dương tính).
- FP (dương tính giả): Khi kết quả xét nghiệm là 1 (dương tính) nhưng kết quả gắn nhãn ban đầu là 0 (âm tính), kết quả của công cụ xét nghiệm không chính xác.
- FN (âm tính giả): Khi kết quả xét nghiệm là 0 (âm tính) nhưng kết quả gắn nhãn ban đầu là 1 (dương tính), kết quả của công cụ xét nghiệm không chính xác.
- TN (âm tính thật): Khi kết quả xét nghiệm và kết quả gắn nhãn trùng khớp nhau đều là 0 (âm tính).

Độ chính xác của công cụ được đánh giá qua các đại lượng precision và recall theo công thức sau:

$$\text{Precision} = \frac{tp}{tp + fp}$$

$$\text{Recall} = \frac{tp}{tp + fn}$$

Công việc kiểm tra độ chính xác của một công cụ xét nghiệm được thực hiện trên máy tính: họ sử dụng 2 tập tin để lưu giá trị: tập tin ghi nhãn "true label.inp" và tập tin ghi kết quả xét nghiệm "predicted label.inp". Hai tập tin chứa dãy số có giá trị 0 hoặc 1 giữa 2 số cách nhau 1 khoảng trắng. Viết chương trình thực hiện:

- Hãy đọc 2 tập tin trên và gán dữ liệu vào 2 mảng được cấp phát động. (1 điểm)
- Tính toán giá trị TP, FP, TN, FN, precision và recall. (2 điểm)
- In kết quả lên tập tin "**result.out**" gồm các thông tin: (1 điểm)
  - Số lượng mẫu xét nghiệm
  - Ma trận Confusion Matrix
  - Giá trị precision và recall.

Ví dụ

STT	0	1	2	3	4	5	6	7	8
true label	0	1	0	1	0	0	0	1	0
predicted label	1	1	0	0	0	0	0	1	1
Kết quả	FP	TP	TN	FN	TN	TN	TN	TP	FP

Ma trận Confusion Matrix

Confusion Matrix		Predicted label	
		1: dương tính	0: âm tính
True label	1: dương tính	TP = 2	FN = 1
	0: âm tính	FP = 2	TN = 4

Tổng số mẫu = 9;

Precision =  $TP / (TP + FP) = 2 / (2 + 2) = 0.5$

Recal =  $TP / (TP + FN) = 2 / (2 + 1) = 0.66$

### Bài 3. (4 điểm)

Văn phòng Khoa ghi nhận danh sách sinh viên. Họ và tên sinh viên được đặt trên mỗi dòng trong một tập tin có tên "**danhsach.inp**", đầu và cuối họ tên không có khoảng trắng, ở giữa mỗi từ có một khoảng trắng. Hãy viết chương trình thực hiện:

1. Đếm xem danh sách có bao nhiêu sinh viên. Xuất số lượng sinh viên ra màn hình. (1 điểm)
2. Chuẩn hóa dữ liệu bằng cách: viết hoa kí tự đầu của mỗi từ trong họ tên sinh viên, các kí tự còn lại viết thường. Danh sách mới được lưu trong tập tin "**Ds\_new.out**". (1 điểm)

Ví dụ: Nguyễn ngọc an, chuyển thành Nguyễn Ngọc An

3. Tìm và in ra màn hình họ tên sinh viên dài nhất (có nhiều kí tự nhất). (1 điểm)  
**Sinh viên chọn 1 trong 2 câu 4 và 5**
4. Để ước tính số lượng cho việc sắp xếp phòng thi, thư kí khoa thực hiện thống kê số lượng sinh viên có **phần tên** bắt đầu cùng một chữ cái. Viết chương trình tính số sinh viên có phần tên bắt đầu cùng 1 chữ cái (họ và tên đã được chuẩn hóa ở câu 2). In kết quả ra màn hình.

Ví dụ :

Danh sách gồm Nguyễn Ngọc An; Phạm Ngọc Anh; Tran Van Minh; Le Van Ha

Chữ cái	A	M	H	...	
Số SV	2	1	1	...	

5. Sắp xếp danh sách sinh viên theo thứ tự tăng dần của phần tên sinh viên. Danh sách được in lên tập tin có tên "**DS\_Sort.out**". (1 điểm)

## YÊU CẦU VỀ CÁCH LÀM BÀI

### 1. về cấu trúc bài kiểm tra

Tất cả các bài sẽ thực hiện trên 1 file được đặt tên theo cấu trúc Ho Ten\_mã số sv.cpp.

Cấu trúc file cpp

Dòng 1: ghi họ và tên sinh viên, mã số sinh viên

Ghi chú thứ tự bài trước code của mỗi bài tương ứng

Phần main() gọi hàm của từng bài.

Ví dụ:

```
//Họ tên: Nguyễn Ngọc An ms 3112000..
```

```
#include...
```

```
//Bai 1
```

Code các Hàm của bài 1

```
//bai 2
```

Code các Hàm của bài 2

```
...
```

```
int main ()
```

```
{
```

```
//Phần kiểm tra bài 1:
```

Khai báo biến

Gọi hàm bài 1

```
//Phần kiểm tra bài 2:
```

Khai báo biến

Gọi hàm bài 2

```
//Phần kiểm tra bài 3:
```

Khai báo biến

Gọi hàm bài 3

```
Return 0;
```

```
}
```

## **2. về yêu cầu viết code**

- Chương trình bắt buộc viết bằng các hàm con.
- Phải có ghi chú ý nghĩa của mỗi hàm.