CANTHO UNIVERSITY

Thực hành

KỸ THUẬT QUY HOẠCH ĐỘNG

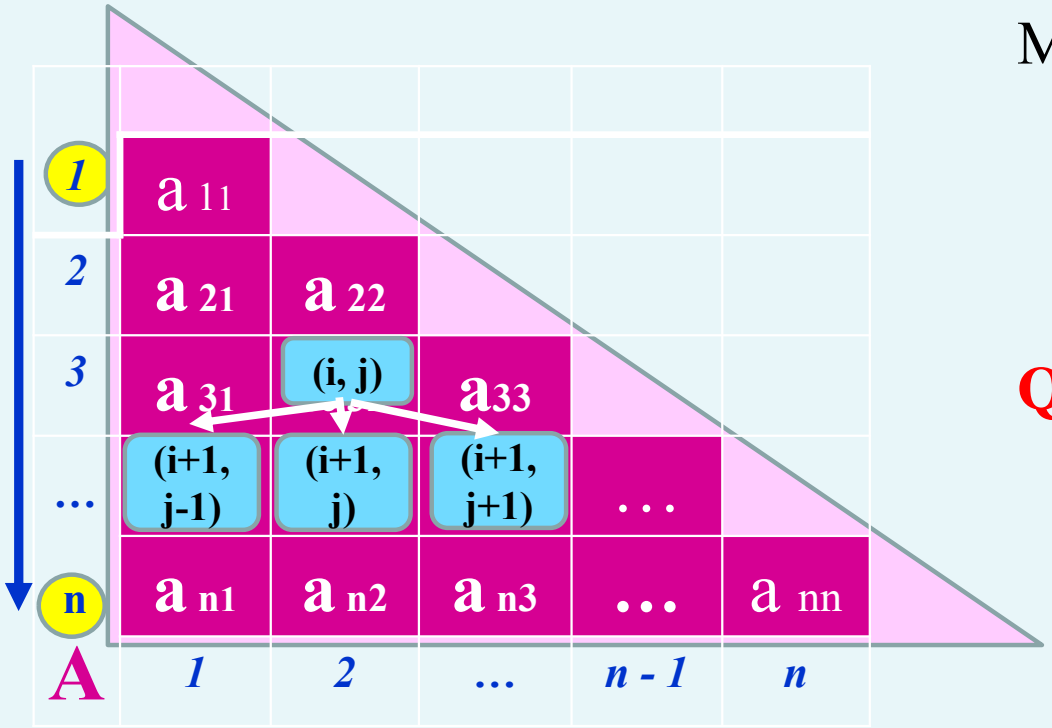
BÀI TOÁN TAM GIÁC SỐ

f(X) → MAX

Võ Huỳnh Trâm

Bài toán TAM GIÁC SỐ

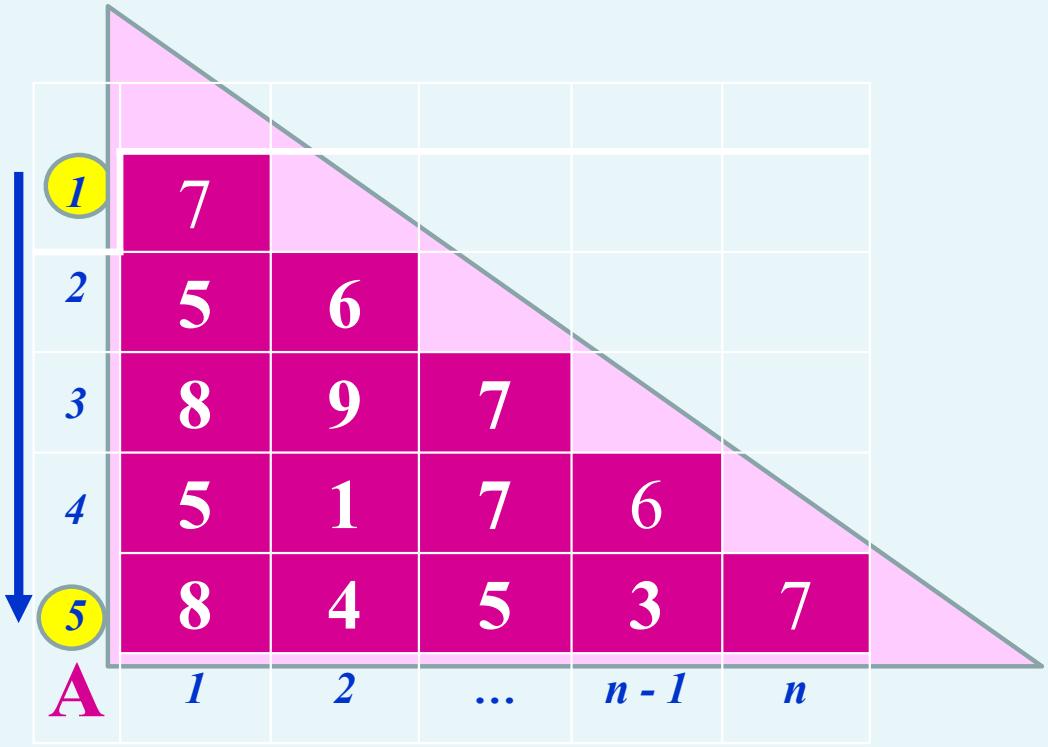
Bài tập: Cho tam giác số kích thước n (ghi các số nguyên) như sau:



Một người xuất phát từ dòng đầu (dòng 1) cần sang dòng cuối cùng (dòng n) sao cho **tổng các số** trên đường đi là **lớn nhất** ?

Quy tắc di chuyển: **từ ô (i, j) có thể di chuyển sang các ô (i+1, j-1), (i+1, j) và (i+1, j+1)**

Bài toán TAM GIÁC SỐ



Ví dụ: Cho tam giác số **A** gồm 5 dòng với các số nguyên được cho như bên.

→ Hãy tìm **hành trình** đi từ dòng 1 sang dòng 5 theo quy tắc di chuyển sao cho tổng giá trị của các số ghi trên đường đi là **lớn nhất** ?

f(X) → Max

Thuật toán QUY HOẠCH ĐỘNG – Xây dựng Công thức truy hồi

(1) Tạo bảng: Xây dựng Công thức truy hồi

- Gọi F**[i, j]** là tổng giá trị các số trên đường đi tới ô a_{11} đến a_{ij}

$\Rightarrow$ F**[n, k]** là tổng giá trị các số trên đường đi tới ô a_{11} đến a_{nk} (phần tử thứ k nào đó trên dòng n)

$\Rightarrow$ F**[n, k] lớn nhất** thì tất cả các F**[i, j]** phải **lớn nhất**

- **Với i = 1** (dòng 1): Vị trí xuất phát là a_{11}

$$\mathbf{F[1,1]} = a_{11}$$

- **Với i = 2** (dòng 2): $\mathbf{F[2,1]} = \mathbf{F[1,1]} + a_{21}$

$$\mathbf{F[2,2]} = \mathbf{F[1,1]} + a_{22}$$

Thuật toán QUY HOẠCH ĐỘNG – Xây dựng Công thức truy hồi

(1) Tạo bảng: Xây dựng Công thức truy hồi

- **Với i > 2** (dòng 3 đến dòng n): 4 trường hợp

- Cột 1: $F[i,1] = Max(F[i-1,1], F[i-1,2]) + a_{i1}$
- Cột i: $F[i,i] = F[i-1,i-1] + a_{ii}$
- Cột i -1: $F[i,i-1] = Max(F[i-1,i-2], F[i-1,i-1]) + a_{ii-1}$
- Cột j ($j \neq 1$, $j \neq i-1$ và $j \neq i$)

$$F[i,j] = Max(F[i-1,j-1], F[i-1,j], F[i-1,j+1]) + a_{ij}$$

TẠO BẢNG: Công thức truy hồi

Công thức truy hồi :

$$F[i, j] = \begin{cases} a_{11} & \text{nếu } i = 1 \\ \text{Max}\,(F[i-1, j-1]; F[i-1, j]; F[i-1,j+1]) + a_{ij} & \text{nếu } i > 1 \end{cases}$$

(với j = 1 .. n)

HÀM XÁC ĐỊNH CHỈ SỐ CỘT MAX

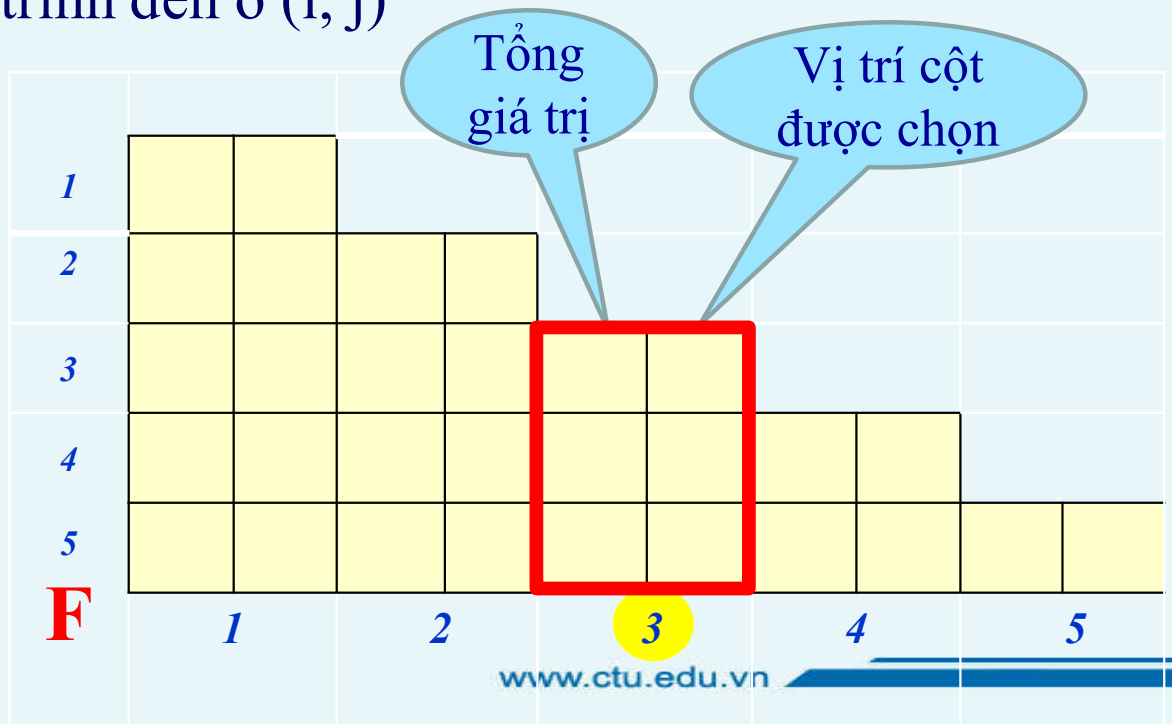
*Hàm trả về chỉ số cột chứa tổng giá trị **lớn nhất** ở dòng i - 1*

```
int CS_max(int F[][size], int i, int j){
   if (j= =0) return (F[i-1][0]>F[i-1][1])? 0:1;
   if (j= =i) return i-1;
   if (j= =i-1)  return (F[i-1][i-2]>F[i-1][i-1])? i-2:i-1;
   if (F[i-1][j-1]>F[i-1][j] && F[i-1][j-1]>F[i-1][j+1])  return j-1;
   if (F[i-1][j]>F[i-1][j-1] && F[i-1][j]>F[i-1][j+1])  return j;
   if (F[i-1][j+1]>F[i-1][j] && F[i-1][j+1]>F[i-1][j-1])  return j+1;
}
```

TẠO BẢNG: Xây dựng bảng F

- Tạo bảng tam giác F có 5 dòng và 5 cột. Mỗi cột chia thành 2 cột con: cột bên trái là ***tổng giá trị***, cột bên phải lưu ***vị trí cột max được chọn.***
- **F(i, j)** là **tổng giá trị** hành trình đến ô (i, j)

1	7				
2	5	6			
3	8	9	7		
4	5	1	7	6	
5	8	4	5	3	7
A	1	2	3	4	5

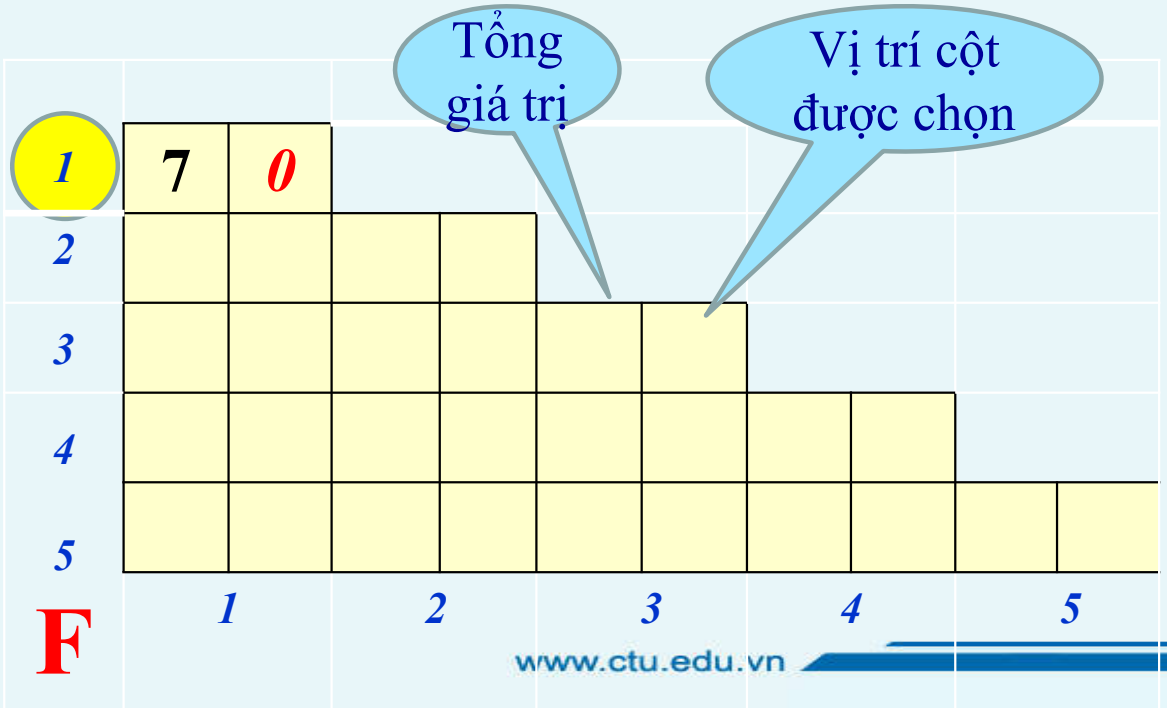


TẠO BẢNG: Xây dựng bảng F

F[1,1] = a_{11} , i = 1

Ví dụ: **F[1,1]** = A[1,1] = 7; không có dòng được chọn trước đó.

1	7				
2	5	6			
3	8	9	7		
4	5	1	7	6	
5	8	4	5	3	7
A	1	2	3	4	5



TẠO BẢNG: Xây dựng bảng F

$$F[i,j] = \mathrm{Max}(F[i-1,j-1], F[i-1,j], F[i-1,j+1]) + a_{ij} \qquad i > 1$$

Ví dụ: **F[5,3]** = F[4,3] + a_{53} = 29 + 5 =34; cột trước đó là 3.

1	7				
2	5	6			
3	8	9	7		
4	5	1	7	6	
5	8	4	5	3	7
A	1	2	3	4	5

Tổng giá trị

Vị trí cột được chọn

1	7	0								
2	12	1	13	1						
3	21	2	22	2	20	2				
4	27	2	23	2	29	2	26	3		
5	35	1	33	3	34	3	32	3	33	4
F	1		2		3		4		5	

HÀM TẠO BẢNG

```
void Tao_Bang(int a[][size], int n, int F[][size]){
 int i, j;
      F[0][0]= a[0][0];
      F[1][0]= a[1][0] + F[0][0];
      F[1][1]= a[1][1] + F[0][0];
        for (i=2; i<n; i++)
          for (j=0; j<=i; j++){
               int k=CS_max(F,i,j);
               F[i][j] = a[i][j]+ F[i-1][k];  } }
```

(2) Tra bảng: Tìm phương án

- Cuối cùng, chọn trên dòng n cột có giá trị **F[n, i]** lớn nhất và từ đó truy ngược lại để tìm hành trình.
- Xét từ dòng cuối n = 5 đến dòng đầu n = 1

TRA BẢNG: TÌM PHƯƠNG ÁN

. n = 5, Max F(**5**,i) = 35, cột trước đó là **1**

. n = 4, chọn F(**4**,**1**) = 27, cột trước đó là **2**

. n = 3, chọn F(**3**,**2**) = 22, cột trước đó là **2**

. n = 2, chọn F(**2**,**2**) = 13, cột trước đó là **1**

. n = 1, chọn F(**1**,**1**) = 7

F	1		2		3		4		5	
1	7	0								
2	12	1	13	1						
3	21	2	22	2	20	2				
4	27	2	23	2	29	2	26	3		
5	35	1	33	3	34	3	32	3	33	4

. Vậy ta có đường đi :

(1,1) → (2,2) → (3,2) → (4,1) → (5,1)

Tổng giá trị = **35**

HÀM XÁC ĐỊNH CỘT MAX CUỐI

```
int CS_max_dong_cuoi (int F[], int j){
        int somax = F[0];
        int maxindex = 0;
        int k;
                for (k=1; k<=j; k++)
                if (F[k] > somax){
                        somax = F[k];
                        maxindex = k;   }
        return maxindex; }
```

*Hàm trả về chỉ số cột chứa tổng giá trị **lớn nhất ở dòng cuối** tam giác*

HÀM TRA BẢNG

```
void Tra_Bang(int a[][size], int n, int F[][size], int PA[]){
        int i,j;
        j = CS_max_dong_cuoi(F[n-1], n-1);
        PA[n-1] = a[n-1][j];
        for (i=n-1; i>=1; i--){
                j = CS_max(F, i, j);
                PA[i-1] = a[i-1][j];
        }
}
```