**Trần Hoàng Triển - MSSV:46.01.104.196**

**Cài đặt bài toán Tháp Hà Nội**

**Dữ liệu vào:**

Một số tự nhiên n là số đĩa cần di chuyển cho trước ở cột A. (1 <= n <= 20)

Ví dụ:

**A picture containing text, font, white, screenshot

Description automatically generated**

**Dữ liêu ra :**

Gồm nhiều dòng là các bước giải bài toán theo cách tối ưu nhất.

Mỗi dòng có cấu trúc "k Src Dest" với:

- k là một số tự nhiên đại diện cho đĩa cần chuyển.

- Src là một chữ cái in hoa đại diện cho tên cột hiện tại của đĩa k.

- Dest là một chữ cái in hoa đại diện cho tên cột cần chuyển đến của đĩa k.

**A screenshot of a computer

Description automatically generated with low confidence**

**Miêu tả cách xử lý dữ liệu như sau:**

* Sau khi nhập `n` là số lượng đĩa cần chuyển.
* Trong hàm `THN`, sử dụng câu điều kiện để xác định khi nào dừng đệ quy:
* Nếu `n` bằng 1, tức chỉ có một đĩa cần chuyển, ta in ra "num Src Dest" để chỉ ra bước chuyển đĩa duy nhất.
* Ngược lại, thực hiện các bước sau:

- Bước 1: Di chuyển `n-1` đĩa từ cột nguồn (`Src`) sang cột trung gian (`Tmp`) bằng cách gọi đệ quy với tham số thích hợp.

- Bước 2: Di chuyển đĩa thứ `num` từ cột nguồn (`Src`) sang cột đích (`Dest`).

- Bước 3: Di chuyển `n-1` đĩa từ cột trung gian (`Tmp`) sang cột đích (`Dest`) bằng cách gọi đệ quy với tham số thích hợp.

3. Trong hàm `main`, nhập vào số lượng đĩa `n`.

4. Gọi hàm `THN` với các tham số ban đầu `n`, `n`, 'A', 'C', 'B' để giải quyết bài toán "Tháp Hà Nội".

5. Kết thúc chương trình in ra là các bước chuyển đĩa theo cách tối ưu nhất để chuyển n đĩa từ cột A sang cột C.

**Để thực thi chương trình trên, bạn có thể làm theo các bước sau:**

1. Mở trình biên dịch C++ (như Dev-C++, Visual Studio,).

2. Tạo một tệp tin mới và sao chép đoạn mã vào tệp tin đó.

3. Lưu tệp tin với phần mở rộng `.cpp` (ví dụ: `**ThapHaNoi.cpp**`).

4. Tiến hành chạy chương trình và nhập dữ liệu từ bàn phím `n` là số lượng đĩa cần chuyển.

5. Trình biên dịch sẽ thực thi chương trình và hiển thị kết quả.

**Source code:**

#include <iostream>

using namespace std;

void THN(int n, int num, char Src, char Dest, char Tmp){

if (n==1) cout <<num<<" "<<Src<<" "<<Dest<<endl;

else{

THN(n-1,n-1,Src,Tmp,Dest);

THN(1,num,Src,Dest,Tmp);

THN(n-1,n-1,Tmp,Dest,Src);

}

}

int main(){

int n; cin>>n;

THN(n,n,'A','C','B');

return 0;

}