**Thuật toán sinh**

1. Khái niệm: Là phương pháp để liệt kê các cấu hình tổ hợp chính tắc của 1 tập hợp.

-Bài toán có thể thực hiện nếu thoả mãn 2 điều kiên sau:

+Có thể **xác định được thứ tự** trên tập hợp các cấu hình tổ hợp cần liệt kê. Từ đó biết được cấu hình đầu tiên và cấu hình cuối cùng.

+**Xây dựng được thuật toán** từ một cấu hình chưa phải cuối, sinh ra được cấu hình kế tiếp.Khi đến cấu hình cuối, kết thúc.

-Tổng quát:

<Xây dựng cấu hình đầu tiên>;

repeat

<Đưa ra cấu hình đang có>;

<Từ cấu hình đang có sinh ra cấu hình tiếp theo nếu còn>;

until <hết cấu hình>;

-Độ phức tạp của thuật toán : Bằng với số lượng cấu hình ta phải sinh ra.

1. Sinh dãy nhị phân:

-Đề bài: Cho 1 số n,hãy sinh ra các số nhị phân có n chứ số.

VD: Với n=3 ta có: 000,001,010,011,100,101,110,111 (8 dãy)

-Cách làm:+Tạo cấu hình đầu tiên : VD: 000

+Sinh các cấu hình tiếp theo với điều kiện:

-Duyệt từ cuối lên đầu. Nếu gặp 1 thì chuyển thành 0,đi tiếp

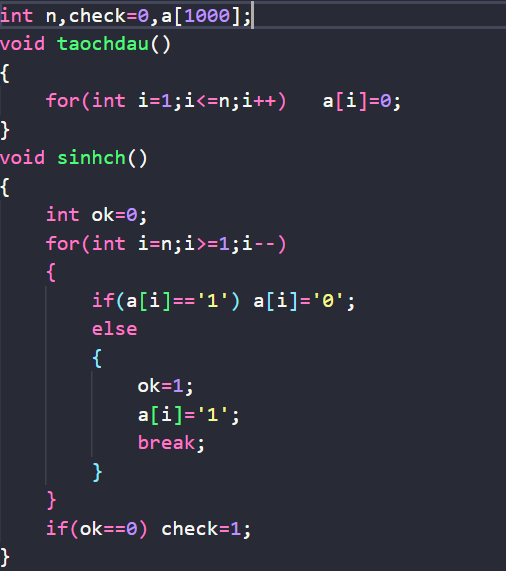
. Nếu gặp 0 thì chuyển thành 1,thoát vòng lặp

-Nếu bước duyệt không tìm thấy số 0 nào

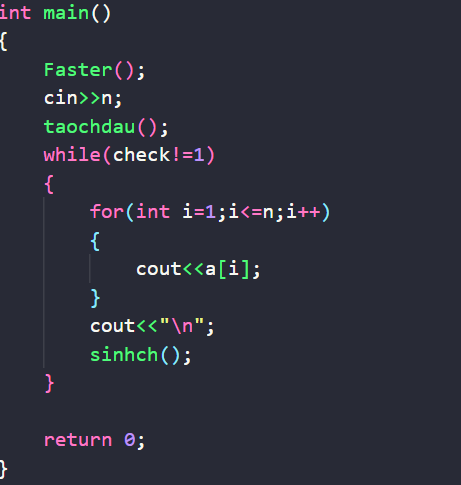
==>Đã đến cấu hình cuối cùng. VD: 111.

Code:[B9Q14n - Online C++ Compiler & Debugging Tool - Ideone.com](https://www.ideone.com/B9Q14n)

1. Tạo cấu hình đầu,sinh các cấu hình tiếp theo:



1. In ra các cấu hình:



1. Sinh Tổ Hợp:

-Đề bài:Liệt kê các xâu con có K phần tử,các phần tử chỉ thuộc đoạn từ 1->N,theo thứ tự tăng dần.

VD:N=5, K=3: 123,124,125,134,135,145,234,235,245,345.

-Cách làm:

+Tạo cấu hình đầu tiên : VD: 123

+Sinh các cấu hình tiếp theo với điều kiện:

-Duyệt từ cuối lên đầu:

-Nếu gặp a[I] <MAX tại vị trí đó(MAX=n-k+i)

thì: a[i]++,các phần tử sau I sẽ tăng dần theo a[I].

VD: 125 có 2<MAX=4 => 2->3,5->4=(3+1) =>Ta có 134.

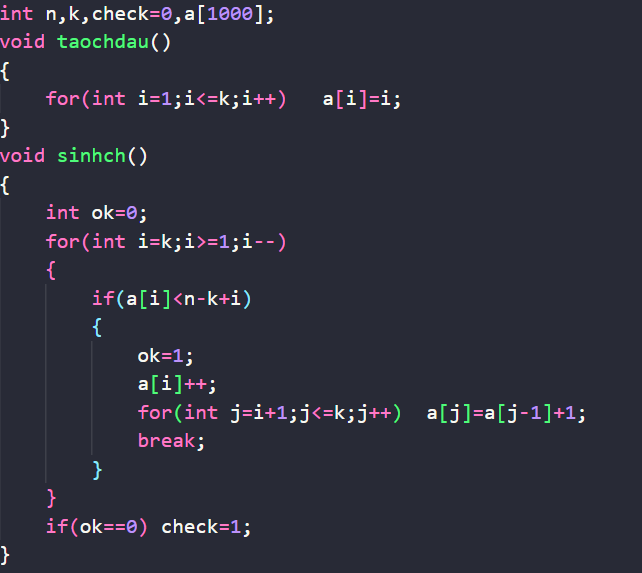
. Nếu không thì đi tiếp

-Nếu bước duyệt không tìm thấy vị trí cần đổi nào: ==>Đã đến cấu hình cuối cùng. VD: 345.

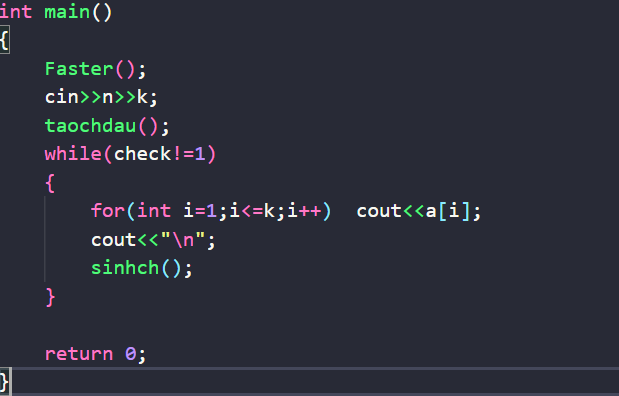
-Độ phức tạp:O(nCk)

Code:[FnjJH7 - Online C++ Compiler & Debugging Tool - Ideone.com](https://www.ideone.com/FnjJH7)

1. Tạo cấu hình đầu,sinh các cấu hình tiếp theo:



2.In ra các cấu hình:



IV.Sinh Hoán Vị:

-Đề Bài:Liệt kê các hoán vị của 1 số.

VD: 123: -> 132,213,231,312,321.

-Cách làm:

+Cấu hình đầu tiên là số đã cho sau khi sắp xếp nó từ bé đến lớn. VD:1234.

+Sinh các cấu hình tiếp theo với điều kiện:

-Duyệt từ cuối lên đầu:

-(\*)Nếu a[I]<a[I+1] :

Đổi chỗ a[i] với phần tử nhỏ nhất đằng sau

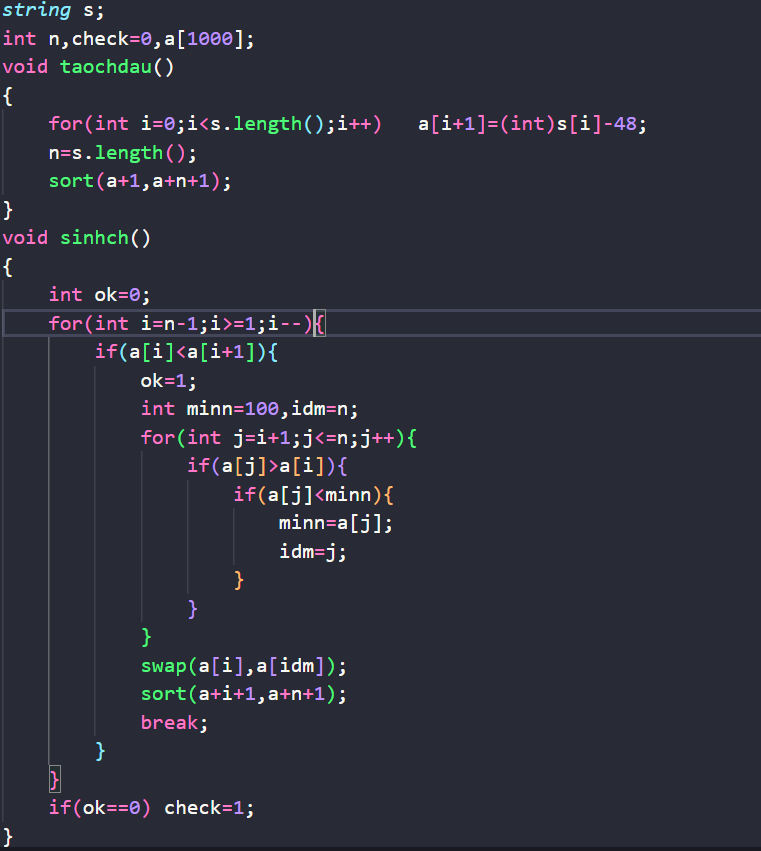
nó,lớn hơn nó.Các chữ số sau khi a[I] sắp xếp lại theo chiều tăng dần VD:1243,đổi 2 cho 3, sắp xếp: =>1324.

-Nếu không thực hiện (\*) lần nào =>Cấu hình cuối

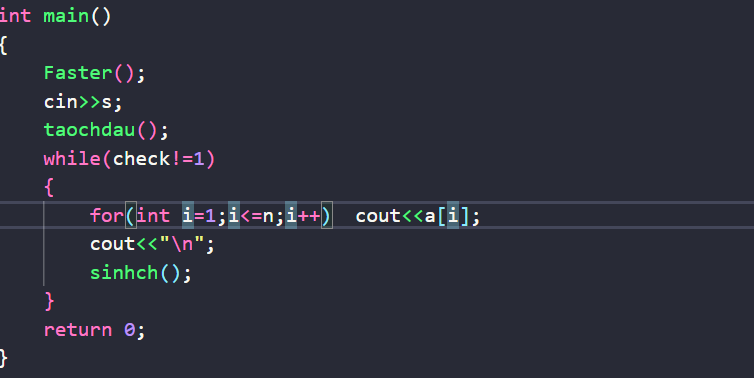
-Độ phức tạp: O(n!)(Nếu tệ nhất là các chữ số đôi một khác nhau)

Code:[V8rx1W - Online C++ Compiler & Debugging Tool - Ideone.com](https://www.ideone.com/V8rx1W)

1.Tạo cấu hình đầu,sinh các cấu hình tiếp theo:



1. In ra các cấu hình:



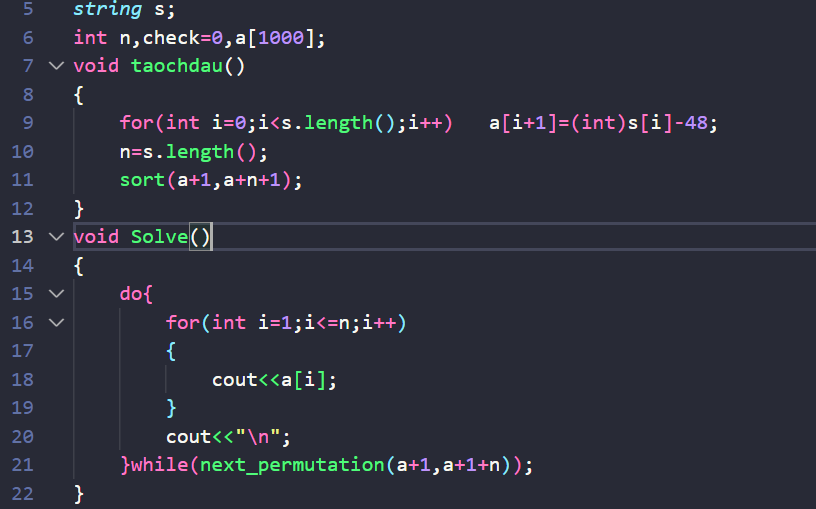
1. Sinh hoán vị bằng hàm có sẵn:(ham bool)

+Next\_premutaion(a+1,a+1+n):In ra từ nhỏ đến lớn.

Khi gọi hàm này,nó sẽ tạo ra ngay cấu hình sau,trả về TRUE(còn ch) hay False(cấu hình cuối(ý là nếu đang là 345,nó sinh ra 1 cấu hình đầu(,bé hơn cấu hình hiện tại),nên nó trả về 0))

+Tối ưu:Khi mà trong dãy có các kí tự giống nhau,việc sinh hoán vị sẽ không in ra các kí tự trùng lặp

Code:[Nvc2fX - Online C++ Compiler & Debugging Tool - Ideone.com](https://www.ideone.com/Nvc2fX)



+Prev\_premutaion(a+1,a+1+n):In ra từ lớn tới nhỏ.

Code:[6iSEfc - Online C++ Compiler & Debugging Tool - Ideone.com](https://www.ideone.com/6iSEfc)

