*Code c++ hay và cần thiết*

Trang format code: <http://www.planetb.ca/syntax-highlight-word>

Mục lục

[***1,Quay lui nhi phân*** 1](#_Toc15380717)

[***2,Liệt kê hoán vị bằng quay lui*** 2](#_Toc15380718)

[***3,liệt kê tổ hợp chập k của n bằng quay lui*** 2](#_Toc15380719)

[***4,Sinh nhị phân*** 3](#_Toc15380720)

[***5,Sinh Tổ hợp chập k của n*** 4](#_Toc15380721)

[***6,Liệt kê hoán vi sinh*** 5](#_Toc15380722)

# ***1,Quay lui nhi phân***

|  |  |
| --- | --- |
| 1. #include<iostream> 3. **using** **namespace** std; 4. **int** n,k,x[200]; 6. **void** output(){ 7. **for**(**int** u=0; u<n; ++u) 8. cout << x[u]; 9. cout << endl; 10. } 12. **void** backTrack(**int** i){ 13. **for**(**int** j=0; j<2; ++j){ 14. x[i] = j; 15. **if** (i==n-1 ) output(); 16. **else** backTrack(i+1); 17. } 18. } 20. **int** main(){ 21. **int** bo;cin>>bo; 22. **while**(bo--){ 23. cin>>n; 25. backTrack(0); 27. } 29. **return** 0; 30. } | In:  -Số bộ test  -Mỗi bộ test nhập vào n  Out:  -Các xấu nhi phân có độ dài n;  J chạy từ 0->1;  Gán lần lượt j vào x[i];  In ra khi I =n-1; |

# ***2,Liệt kê hoán vị bằng quay lui***

|  |  |
| --- | --- |
| 1. //liet ke hoan vi bang quay lui 2. #include<iostream> 4. **using** **namespace** std; 5. **int** n,k,x[200]; 6. **bool** chuaxet[200]={};  //mac dinh chuaxet[i]=false 7. **void** out(){ 8. **for**(**int** i=1;i<=n;++i){ 9. cout<<x[i]<<" "; 10. } 11. cout<<endl; 12. } 13. **void** hoanvi(**int** i){ 14. **for**(**int** j=1;j<=n;++j){ 15. **if**(chuaxet[j]==**false**){ 16. x[i]=j; 17. chuaxet[j]=**true**; 18. **if**(i==n) out(); 19. **else** hoanvi(i+1); 20. chuaxet[j]=**false**; 21. } 22. } 23. } 24. **int** main(){ 25. **int** bo;cin>>bo; 26. **while**(bo--){ 27. cin>>n; 28. //for(int i=1;i<=n;++i) chuaxet[i]=true; 29. hoanvi(1); 30. } 32. **return** 0; 33. } | Mảng chuaxet dùng để đánh dấu phần tử đã xuât hiện trong mang x hay chưa.  Nhưng mặc định tất cả các phần tử của chuaxet là false nên cần cài đặt cho mang này giá trị ban đầu là true  Chú ý :nếu để điều kiện là i==k thì in ra ,ta cũng không được tổ hợp chập k của n vì có lặp. |

# ***3,liệt kê tổ hợp chập k của n bằng quay lui***

|  |  |
| --- | --- |
| 1. //liet ke to hop chap k cua n 2. #include<iostream> 4. **using** **namespace** std; 5. **int** n,k,x[200]; 7. **void** out(){ 8. **for**(**int** i=1;i<=k;++i){ 9. cout<<x[i]<<" "; 11. } 12. cout<<endl; 13. } 14. **void** tohop(**int** i){ 15. **for**(**int** j=x[i-1]+1;j<=n-k+i;j++){ 16. x[i]=j; 17. **if**(i==k) out(); 18. **else** tohop(i+1); 19. } 20. } 22. **int** main(){ 23. **int** bo;cin>>bo; 24. **while**(bo--){ 25. cin>>n>>k; 26. x[0]=0;, 27. tohop(1); 28. } 30. **return** 0; 31. } | Hàm quay lui tohop();  Duyệt j từ x[i-1] +1,đến n-k+I :  Vd n=5,k=3 i=1 ->j=0+1->3;  Mỗi vị trí của x[i] :có phần tử đầu tiên lấy ở tổ hợp đầu tiên ,phần tử cuối lấy ở cấu hình cuối |

# ***4,Sinh nhị phân***

|  |  |
| --- | --- |
| 1. //sinh nhi phan 2. #include<iostream> 4. **using** **namespace** std; 5. **int** x[200]={},n; 6. **bool** ok=**true**; 7. **void** in(){ 8. cin>>n; 9. **for**(**int** i=0;i<=n;++i) x[i]=0; 10. } 11. **void** out(){ 12. **for**(**int** i=1;i<=n;++i) 13. cout<<x[i]<<" "; 14. cout<<endl; 15. } 16. **void** sinh(){ 17. **int** i=n; 18. **while**(i>0 && x[i]==1) { 19. x[i]=0; 20. i--; 21. } 22. **if**(x[i]==0 && i>0) x[i]=1; 23. **else** ok=**false**; 24. } 25. **int** main(){ 26. in(); 27. **while**(ok){ 28. out(); 29. sinh(); 30. } 32. **return** 0; 33. } | Duyệt từ cuối mảng ,nếu xi =1 thì gán x[i]=0 cho đến khi thấy x[i]=0 hoặc hết chuỗi thì dừng lại;  Nếu i>0 thì gán x[i]=1 kết thúc hàm  Nếu khác gán ok=false tức là cấu hình cuối cùng |

# ***5,Sinh Tổ hợp chập k của n***

|  |  |
| --- | --- |
| 1. //sinh to hop chap k cua n 2. #include<iostream> 4. **using** **namespace** std; 5. **int** n,k,x[200]={}; 6. **bool** ok =**true**; 7. **void** init(){ 8. cin>>n; 9. cin>>k; 10. **for**(**int** i=1;i<=k;++i){ 11. x[i]=i; 12. } 13. } 14. **void** out(){ 15. **for**(**int** i=1;i<=k;++i){ 16. cout<<x[i]<<" "; 17. } 18. cout<<endl; 19. } 20. **void** tohop(){ 21. **int** i=k; 22. **while**(i>0 && x[i]==n-k+i) i--; 23. **if**(i>0) { 24. x[i]=x[i]+1; 25. **for**(**int** j=i+1;j<=k;j++) x[j]=x[i]+j-i; 26. } 27. **else** ok=**false**; 28. } 29. **int** main(){ 30. init()  ; 31. **while**(ok){ 32. out(); 33. tohop(); 34. } 35. **return** 0; 36. } |  |

# ***6,Liệt kê hoán vi sinh***

|  |  |
| --- | --- |
| 1. //sinh hoan vi ke tiep 3. #include<iostream> 5. **using** **namespace** std; 6. **int** n,k,x[200]={}; 7. **bool** ok=**true**; 8. **void** ini(){ 9. cin>>n; 10. **for**(**int** i=1;i<n;i++){ 11. x[i]=i; 12. } 13. } 14. **void** out(){ 15. **for**(**int** i=1;i<=n;++i){ 16. cout<<x[i]<<" "; 17. } 18. cout<<endl; 19. } 20. **void** shoanvi(){ 21. **int** j=n-1; 22. **while**(j>0 && x[j]>x[j+1]) j--; 23. **if**(j>0){ 24. **int** k=n; 25. **while**(x[j]>x[k]) k--; 26. swap(x[j],x[k]); 27. **int** r=j+1,s=n; 28. **while**(r<s){ 29. swap(x[r],x[s]); 30. r++;s--; 31. } 32. } 33. **else** ok =**false**; 34. } 36. **int** main(){ 37. ini(); 38. **while**(ok){ 39. out(); 40. shoanvi(); 41. } 42. **return** 0; 43. } | Liệt kê hoán vị bằng phương pháp sinh :  Tìm j là vị trí không đúng |