***QUY HOẠCH ĐỘNG***

1. ***#include <bits/stdc++.h>***
2. ***using namespace std;***

|  |  |
| --- | --- |
| **Bài toán đổi tiền**  int main(){  int n, q;  cin >> n >> q;  int tien[n+2];  vector<int> vt(10001, 10002);  for (int i=0; i<n; i++){  cin >> tien[i];  vt[tien[i]]=1;  }  sort(tien, tien + n);  for (int i=0; i<10001; i++){  for (int j=0; j<n; j++){  if (i>tien[j]){  vt[i] = min(vt[i], vt[i-tien[j]]+1);  }  else  break;  }  }  int truyVan;  while (q--){  cin >> truyVan;  if (vt[truyVan]<10002)  cout << vt[truyVan] << endl;  else  cout << -1 << endl;  }  } | **Sắp xếp balo**  int balo[102][10002] = {};  int n, q ,w[102], v[102];  int main(){  cin >> n;  for (int i=1; i<=n; i++)  cin >> w[i] >> v[i];  for (int i=1; i<=n; i++)  {  for (int j=1; j<=10001; j++)  {  if (j<w[i])  balo[i][j] = balo[i-1][j];  else  balo[i][j] = max(balo[i-1][j], balo[i-1][j-w[i]] + v[i]);  }  }  cin >> q;  while(q--)  {  int test;  cin >> test;  cout << balo[n][test] << endl;  }  } |
| **Bảng số**  int main()  {  int n, m;  cin >> n >> m;  int a[n+5][m+5];  for (int i=0; i<n; i++)  for (int j=0; j<m; j++)  cin >> a[i][j];  for(int i=1; i<n; i++)  a[i][0] += a[i-1][0];  for(int j=1; j<m; j++)  a[0][j] += a[0][j-1];  for (int i=1; i<n; i++)  for (int j=1; j<m; j++)  {  a[i][j] += max(a[i-1][j], a[i][j-1]);  }  cout << a[n-1][m-1];  } | **Tìm tất cả những xâu con chung dài nhất**  **s**tring s1, s2;  int l1, l2, LCSLength = 0;  int C[MAX][MAX];  int calcLCSLenght(int i, int j)  {  int &lcsLen = C[i][j];  if (i==l1 || j==l2) return lcsLen = 0;  if (lcsLen != -1) return lcsLen;  lcsLen = 0;  if (s1[i] == s2[j]) lcsLen = 1 + calcLCSLenght(i+1, j+1);  else lcsLen = max(calcLCSLenght(i+1, j), calcLCSLenght(i, j+1));  return lcsLen;  }  void printAllLCS(char data[], int u, int v, int c) {  if (c == LCSLength)  {  data[c] = '\0';  puts(data);  return;  }  if (u == l1 || v == l2) return;  for (char x = 'a'; x <= 'z'; x++)  {  bool check = false;  for (int i=u; i<l1; i++)  {  if (x==s1[i])  {  for (int j=v; j<l2; j++)  {  if (x==s2[j] && calcLCSLenght(i, j) == LCSLength-c)  {  data[c] = x;  printAllLCS(data, i+1, j+1, c+1);  check = true;  break;  }  }  }  if(check) break;  }  }  }  int main(){  cin >> s1 >> s2;  l1 = s1.length();  l2 = s2.length();  memset(C, -1, sizeof(C));  LCSLength = calcLCSLenght(0, 0);  if(!LCSLength) {cout << "khong co xau con chung"; return 0;}  char data[MAX];  printAllLCS(data, 0, 0, 0); return 0;**}** |

***QUAY LUI***

|  |  |
| --- | --- |
| **Đổi tiền**  int n, q, tien[102], res;  void quayLui(int test, int soTo)  {  if (test==0){  if (soTo<res)  res = soTo;  return;  }  for (int i=0; i<n; i++)  {  if (tien[i]>test)  return;  else{ //test = to tien[i] + doi test-tien[i]  quayLui(test-tien[i], soTo+1);  }  }  }  void inKQ(){  if (res<1e4+1)  cout << res << endl;  else  cout << -1;  }  int main()  {  cin >> n >> q;  for (int i=0; i<n; i++)  cin >> tien[i];  sort(tien, tien+n);  for (int i=0; i<q; i++)  {  int test;  cin >> test;  res = 1e4+1;  quayLui(test, 0);  inKQ();  }  }; | **Sắp xếp ba lô**  int n, q, res=0;  pair<int, int> p[102];  void Try(const int &test, const int &k, const int &m)  {  for (int i=k; i<n; i++)  {  if (p[i].f>test)  return;  else  {  res = max(res, m+p[i].s);  Try(test-p[i].f, i+1, m+p[i].s);  }  }  }  int main(){  ios::sync\_with\_stdio(0); cin.tie(0); cout.tie(0);  cin >> n;  for (int i=0; i<n; i++)  cin >> p[i].f >> p[i].s;  sort(p, p+n);  cin >> q; // so test  while(q--)  {  int test;  cin >> test; res = 0;  Try(test, 0, 0);  cout << res << endl;;  }  } |
| **Người đi du lịch**   1. **int n, res = INT\_MAX;** 2. **int kt[16] = {};// danh dau xem tp i da den chua** 3. **int hVi[16]; // hoan vi** 4. **int kCach[16][16];** 5. **void Try(int cnt){** 6. **if(cnt == n){** 7. **int sum = kCach[hVi[0]][hVi[n-1]];** 8. **for (int i=0; i<n-1; i++)** 9. **sum += kCach[hVi[i]][hVi[i+1]];** 10. **if (sum<res)** 11. **res = sum;** 12. **return;** 13. **}** 14. **for (int i=0; i<n; i++){** 15. **if (!kt[i])** 16. **{** 17. **hVi[cnt] = i;** 18. **kt[i]=1;** 19. **Try(cnt+1);** 20. **kt[i]=0;** 21. **}** 22. **}** 23. **}** 24. **int main()** 25. **{** 26. **cin >> n;** 27. **for (int i=0; i<n; i++)** 28. **for (int j=0; j<n; j++)** 29. **cin >> kCach[i][j];** 30. **Try(0);** 31. **cout << res;** 32. **}** | **Tám hậu**  int res, n;  //ko dc an nhau khi ko trung hang, cot, duong cheo  int cheoPhai[100], hang[100], cheoTrai[100];  void Try(int k)  {  if (k==n)  {  res ++;  return;  }  for (int i=1; i<=n; i++)  {  if (!hang[i]&&!cheoPhai[i+k]&&!cheoTrai[i-k+50])  {  hang[i]=cheoPhai[i+k]=cheoTrai[i-k+50]=1;  Try(k+1);  hang[i]=cheoPhai[i+k]=cheoTrai[i-k+50]=0;  }  }  }  int main()  {  cin >> n;  cheoPhai[n] = 1;  cheoTrai[1+50] = 1;  Try(0);  cout << res;  } |
| **Hoán vị lặp.**  int len;  //mang danh dau, ki tu thu i da duoc su dung chua  bool ktra[14];  string s, tmp;  map<string, bool> mp;  void Try()  {  if (tmp.size() == len)  {  mp[tmp]=1;  return;  }  for (int i=0; i<len; i++)  {  if(!ktra[i])  {  ktra[i] = 1;  tmp += s[i];  Try();  ktra[i] = 0;  tmp.pop\_back();  }  }  }  int main()  {  ios::sync\_with\_stdio(0);cin.tie(0);cout.tie(0);  cin >> s;  len=s.size();  Try();  for(auto x:mp)  cout << x.first << endl;  } | **Phân tích số (thành tổng một dãy không giảm các số nguyên dương).**  int cnt = 0;  void Try(int n, int k)  {  if (n==0)  {  cnt ++;  return;  }  for (int i=k; i<=n; i++)  Try(n-i, i);  }  void sol()  {  int n;  cnt = 0;  cin >> n;  Try(n, 1);  cout << cnt << endl;  }  int main()  {  ios::sync\_with\_stdio(0);cin.tie(0);cout.tie(0);  int t;  cin >> t;  while (t--)  sol();  } |

***CHIA ĐỂ TRỊ***

|  |  |
| --- | --- |
| **xâu Fibonacci.**  typedef long long ll;  typedef unsigned long long ull;  ll f[94];  void khoiTao()  {  f[1]=1; f[2]=1;  for (int i=2; i<94; i++)  f[i] = f[i-2] + f[i-1];  }  char xauFibo(int n, ll k)  {  if (k>f[n])  return '$';  if (n==1)  return 'A';  if (n==2)  return 'B';  if (k<=f[n-2])  return xauFibo(n-2, k);  else  return xauFibo(n-1, k-f[n-2]);  }  int main()  {  ios::sync\_with\_stdio(0);cin.tie(0);cout.tie(0);  int t, n;  ll k;  khoiTao();  cin >> t;  while (t--)  {  cin >> n >> k;  cout << xauFibo(n, k) << endl;  }  } | **Cắt thanh kim loại.**  struct Tree  {  int a,b,m;  Tree \*Left,\*Right;  Tree(int u,int v)  {  a=u;  b=v;  m=b-a;  Left=Right=NULL;  }  };  void add(Tree \*T,int x)  {  if(T->Left==0)  {  T->Left=new Tree(T->a,x);  T->Right=new Tree(x,T->b);  }  else  {  if(x<T->Left->b) add(T->Left,x);  else add(T->Right,x);  }  T->m=max(T->Left->m,T->Right->m);  }  int main()  {  int n,m,x;  scanf("%d%d",&n,&m);  Tree \*T=new Tree(0,m);  for(int i=1;i<=n;i++)  {  scanf("%d",&x);  add(T,x);  printf("%d ",T->m);  }  } |
| **Đếm số nghịch thế (cây IT).**  typedef long long ll;  typedef unsigned long long ull;  ll res = 0;  struct node  {  int canTrai, canPhai, elem; //quan ly doan nao elem cua doan  node \*L, \*R;  node(int u, int v)  {  canTrai = u, canPhai = v;  elem = 0;  int mid = (u+v)/2;  if (u==v)  L = R =NULL;  else  {  L = new node(u, mid);  R = new node(mid+1, v);  }  }  };  //dua a[k] = x vao cay  void update(node \*&N, int k, int x)  {  N->elem++;  if (N->L)  if(x<=N->L->canPhai)  {  res+= N->R->elem;  return update(N->L, k, x);  }  else  return update(N->R, k, x);  }  int main()  {  ios::sync\_with\_stdio(0);cin.tie(0);cout.tie(0);  int n,tmp;  cin >> n;  node \*root = new node(1, n);  for (int i=1; i<=n; i++)  {  cin >> tmp;  update(root, i, tmp);  }  cout << res;  } | **Đếm số nghịch thế (dựa mergesort).**  int b[100006],c[100006];  long long res=0;  void ms(int \*a,int L,int R) //sx a[L]..a[R]  {  if(L>=R) return;  int M=(L+R)/2;  ms(a,L,M);  ms(a,M+1,R);  int i=0,j=0;  for(int k=M;k>=L;k--) b[++i]=a[k];  for(int k=R;k> M;k--) c[++j]=a[k];  for(int k=L;k<=R;k++)  if(i>0 && j>0)  {  if(c[j]<b[i]) {res+=i; a[k]=c[j--];}  else a[k]=b[i--];  }  else a[k]=i>0?b[i--]:c[j--];  }  int main()  {  int n;  cin>>n;  int a[n+5];  for(int i=1;i<=n;i++) cin>>a[i];  ms(a,1,n);  cout<<res;  } |

***Name: Tim phan tu lon nhat - nho nhat trong mang bang De quy***  
***#include <conio.h>  
#include<stdio.h>  
  
int maxc(int a, int b){  
    return a>b?a:b;  
}  
  
int minc(int a, int b){  
    return a<b?a:b;  
}  
int searchmax(int a[], int n){  
  
  if (n==0) return a[0];  
  else  return maxc(a[n-1],searchmax(a,n-1));  
}  
int searchmin(int a[], int n){  
    if(n==0) return a[0];  
    else return minc(a[n-1],searchmin(a,n-1));  
}  
main(){  
 int n=5;  
    int a[n]={1,5,10,9,8};  
    printf("\t Gia tri lon nhat: %d",searchmax(a,n));  
    printf("\n\t Gia tri nho nhat: %d",searchmin(a,n));  
  
    getch();  
    return 0;  
}***