|  |  |
| --- | --- |
| #include "stdio.h"  #include "windows.h"  #include "conio.h"  #define VOCUNG 9999  #define DADUYET -2  #define DINH 20  #include "time.h"  #include "stdlib.h"  void docFile(char \*fileName, int a[][DINH], int &n);  void xuat(int a[][DINH], int n);  void KT\_DonDoThi(int a[][DINH], int n);  void KT\_DaDoThi(int a[][DINH], int n);  void KT\_Gia\_Do\_Thi(int a[][DINH], int n);  void tenDoThi(int a[][DINH], int n);  void kiemTraBac(int a[][DINH], int n);  void DFS(int a[][DINH], int n, int num, bool mask[]);  void BFS(int a[][DINH], int n, int num);  void duongDiNganNhat(int a[][DINH], int n, int mangddnn[][DINH], int dinh);  void xuatDinhDang(int mangddnn[][DINH], int n, int dinh);  void ddnn\_ThuatToanFloyd(int a[][DINH], int n,int dinh);  void chuTrinhEuler(int a[][DINH], int n);  //---ham phu----  int KT\_DTVoHuong(int a[][DINH], int n);  int KT\_Don\_VoHuong(int a[][DINH], int n);  int KT\_Don\_CoHuong(int a[][DINH], int n);  int KT\_Da\_VoHuong(int a[][DINH], int n);  int KT\_Da\_CoHuong(int a[][DINH], int n);  bool laGiaDoThi(int a[][DINH], int n);  char convert(int a);  int convert(char a);  void doiMangRaPhiaTruoc(int hangdoi[],int &phanTuHangDoi);  int chiSoCoGiaTriMin( int mangddnn[][DINH], int n);  void xuatMangTam(int tam[], int n);  bool conDinh(int a[][DINH], int n, int dinh);  bool coChuTrinhEuler(int a[][DINH], int n);  bool coDuongDiEuler(int a[][DINH], int n);  void menu(int &lc, int a[][DINH], int n)  {  for(;;)  {  system("cls");  printf("\n1.Xuat Ma Tran");  printf("\n2.Kiem Tra Don Do Thi");  printf("\n3.Kiem Tra Da Do Thi");  printf("\n4.Kiem Tra Gia Do Thi");  getch();  }  else if(lc==6)  {  system("cls");  kiemTraBac(a, n);    printf("\n Nhap phim bat ky de tiep tuc!");  getch();  }  else if(lc==7)  {  system("cls");  bool mask[DINH];  int dinh;  for(int i=1; i<=n; i++)  {  mask[i]==false;  }  do{  printf("\nNhap dinh can duyet: ");  scanf("%d", &dinh);  if(dinh < 1 || dinh > n) printf("Loi so dinh toi da la: %d\n", n);  }while(dinh < 1 || dinh > n);    printf("\n=>DFS(%c) = ", dinh);  DFS(a, n, dinh, mask);  printf("\n Nhap phim bat ky de tiep tuc!");  getch();  }  else if(lc==8)  {  system("cls");  bool mask[DINH];  int dinh;  for(int i=1; i<=n; i++)  {  mask[i]==false;  }  do{  printf("\nNhap dinh can duyet: ");  scanf("%d", &dinh);  if(dinh < 1 || dinh > n) printf("Loi so dinh toi da la: %d\n", n);  }while(dinh < 1 || dinh > n);    }  else break;  }  }  int main()  {  int a[DINH][DINH], n, lc;  char fileName[20]="content.txt";  docFile(fileName, a, n);    menu(lc, a ,n);    return 0;  }  void docFile(char \*fileName, int a[][DINH], int &n)  {  FILE \*f = fopen(fileName, "rt");  if(f==NULL)  {  printf("\nChua doc duoc file!");  }  else  {  fscanf(f, "%d", &n);  for(int i=1; i<=n; i++)  {  for(int j=1; j<=n; j++)  {  fscanf(f, "%d", &a[i][j]);  }  }  }    fclose(f);  }  void xuat(int a[][DINH], int n)  {  for(int i=1; i<=n; i++)  {  for(int j=1; j<=n; j++)  {  if(a[i][j] == 9999)  printf("%6s", "oo");  else  printf("%6d", a[i][j]);  }  printf("\n");  }  }  int KT\_Don\_CoHuong(int a[][DINH], int n)  {  for(int i=1; i<=n; i++)  {  for(int j=1; j<=n; j++)  {  if(a[i][j]>1)  return 0;  }  }  return 1;  }  void KT\_DaDoThi(int a[][DINH], int n)  {  printf("\nMa Tran Ke:\n");  xuat(a, n);  if(KT\_DTVoHuong(a, n) == 1)  {  if(KT\_Da\_VoHuong(a, n)==1)  {  printf("\nYES");  }  else printf("\nNO");  }  else  {  if(KT\_Da\_CoHuong(a, n))  {  printf("\nYES");  }  else printf("\nNO");  }  }  int KT\_Da\_VoHuong(int a[][DINH], int n)  {  int dem=0;  for(int i=1; i<=n; i++)  {  for(int j=1; j<=n; j++)  {  if(i==j)  {  if(a[i][j]!=0)  return 0;  }  else if(a[i][j]>=2)  dem++;  }  }  if(dem>=1)  return 1;  else return 0;  }  void tenDoThi(int a[][DINH], int n)  {  printf("\nMa Tran Ke:\n");  xuat(a, n);  if(KT\_DTVoHuong(a, n)==1)  {  if(laGiaDoThi(a, n)==true)  printf("\n => Gia Do Thi!");  else if(KT\_Don\_VoHuong(a, n)==1)  printf("\n => Don Do Thi Vo Huong!");  else printf("\n => Da Do Thi Vo Huong!");  }  else  {  if(KT\_Da\_CoHuong(a, n)==1)  printf("\n => Da Do Thi Co Huong!");  else printf("\n => Don Do Thi Co Huong!");  }  }  void kiemTraBac(int a[][DINH], int n)  {  printf("\nMa Tran Ke:\n");  xuat(a, n);  int bac, bacVao, bacRa;    do{  printf("\nNhap vao bac: ");  scanf("%d", &bac);  }while(bac<=0 || bac>n);  if(KT\_DTVoHuong(a, n)==1)  {  bacVao=0;  for(int i=1; i<=n; i++)  {  if(i==bac)  bacVao += 2 \* a[bac][i];  else bacVao += a[bac][i];  }  printf("\nBac(%d) = %d",bac, bacVao);  }  else  {  bacVao=0; bacRa=0;  for(int i=1; i<=n; i++)  {  bacVao += a[i][bac];  bacRa += a[bac][i];  }  printf("\nBac\_Vao(%d) = %d\nBac\_Ra(%d) = %d",bac, bacVao, bac, bacRa);  }  }  char convert(int a)  {  switch(a)  { case 1: return 'a';  case 2: return 'b';  case 3: return 'c';  case 4: return 'd';  case 5: return 'e';  case 6: return 'f';  case 7: return 'g';  case 8: return 'h';  case 9: return 'k';  case 10: return 'l';  }  }  int convert(char a)  {  switch(a)  { case 'a': return 1;  case 'b': return 2;  case 'c': return 3;  case 'd': return 4;  case 'e': return 5;  case 'f': return 6;  case 'g': return 7;  case 'h': return 8;  case 'k': return 9;  case 'l': return 10;  }  }  int chiSoCoGiaTriMin( int mangddnn[][DINH], int n)  {  int min=0;  for(int i=1; i<=n; i++)  {  if(mangddnn[3][i] != DADUYET)  {  if(min==0)  {  min = i;  }  else  {  if(mangddnn[1][i] < mangddnn[1][min])  {  min = i;  }  }  }  }  return min;  }  void xuatDinhDang(int mangddnn[][DINH], int n, int dinh)  {    printf("\nDuong di ngan nhat tu %c", convert(dinh));  for(int i=1; i<=n; i++)  {    if(i != dinh)  {  int tam[DINH], sl=0;  printf("\n\tDen %c (do dai: %3d) la: ", convert(i), mangddnn[1][i]);  if(mangddnn[1][i]==VOCUNG) printf("Khong co duong di");  else  {  //printf("%2c - > " convert(dinh));  int d = i;  while(mangddnn[2][d] != dinh)  {  sl++;  tam[sl] = mangddnn[2][d];    if(mangddnn[2][d] != dinh)  {  d=mangddnn[2][d];  }  }  printf("%3c ->", convert(dinh));  xuatMangTam(tam, sl);  printf("%3c", convert(i));  }  }  }  }  void xuatMangTam(int tam[], int n)  {  for(int i=n; i>0; i--)  {  printf("%3c ->", convert(tam[i]));  }  } | printf("\n5.Cho Biet Ten Do Thi");  printf("\n6.Bac Cua Dinh");  printf("\n7.Duyet DFS");  printf("\n8.Duyet BFS");  printf("\n9.Duong di ngan nhat");  printf("\n10.Duong di ngan nhat Thuat Toan Floyd");  printf("\n11.Chu Trinh Euler");  printf("\nToi Muon: ");  scanf("%d", &lc);  if(lc==1)  {  system("cls");  xuat(a, n);    printf("\n Nhap phim bat ky de tiep tuc!");  getch();  }  else if(lc==2)  {  system("cls");  KT\_DonDoThi(a, n);    printf("\n Nhap phim bat ky de tiep tuc!");  getch();  }  else if(lc==3)  {  system("cls");  KT\_DaDoThi(a, n);    printf("\n Nhap phim bat ky de tiep tuc!");  getch();  }  else if(lc==4)  {  system("cls");  KT\_Gia\_Do\_Thi(a, n);    printf("\n Nhap phim bat ky de tiep tuc!");  getch();  }  else if(lc==5)  {  system("cls");  tenDoThi(a, n);    printf("\n Nhap phim bat ky de tiep tuc!");  printf("\n=>BFS(%d-%c) = ", dinh, convert(dinh));  BFS(a, n, dinh);  printf("\n Nhap phim bat ky de tiep tuc!");  getch();  }  else if(lc==9)  {  system("cls");    int mangddnn[4][DINH], dinh;    do  {  printf("\nNhap dinh: ");  scanf("%d", &dinh);  }while(dinh <=0 || dinh >n);  duongDiNganNhat( a, n, mangddnn, dinh);  xuatDinhDang(mangddnn, n, dinh);    printf("\n Nhap phim bat ky de tiep tuc!");  getch();  }  else if(lc==10)  {  system("cls");  int dinh;  do  {  printf("\nNhap dinh: ");  scanf("%d", &dinh);  }while(dinh <=0 || dinh >n);  ddnn\_ThuatToanFloyd(a, n, dinh);    printf("\n Nhap phim bat ky de tiep tuc!");  getch();  }  else if(lc==11)  {  system("cls");    chuTrinhEuler(a, n);    printf("\n Nhap phim bat ky de tiep tuc!");  getch();  void KT\_DonDoThi(int a[][DINH], int n)  {  printf("\nMa Tran Ke:\n");  xuat(a, n);  if(KT\_DTVoHuong(a, n) == 1)  {  if(KT\_Don\_VoHuong(a, n)==1)  {  printf("\nYES");  }  else printf("\nNO");  }  else  {  if(KT\_Don\_CoHuong(a, n))  {  printf("\nYES");  }  else printf("\nNO");  }  }  int KT\_DTVoHuong(int a[][DINH], int n)  {  for(int i=1; i<=n; i++)  {  for(int j=i+1; j<=n; j++)  {  if(a[i][j]!=a[j][i])  return 0;  }  }  return 1;  }  int KT\_Don\_VoHuong(int a[][DINH], int n)  {  for(int i=1; i<=n; i++)  {  for(int j=1; j<=n; j++)  {  if(i==j)  {  if(a[i][j]!=0)  return 0;  }  else if(a[i][j]>1)  return 0;  }  }  return 1;  }  int KT\_Da\_CoHuong(int a[][DINH], int n)  {  int dem=0;  for(int i=1; i<=n; i++)  {  for(int j=1; j<=n; j++)  {  if(a[i][j]>1)  dem++;  }  }  if(dem>=1)  return 1;  else return 0;  }  bool laGiaDoThi(int a[][DINH], int n)  {  int dem=0;  for(int i=1; i<=n; i++)  {  for(int j=1; j<=n; j++)  {  if(i==j && a[i][j]>0)  dem++;  }  }  if(dem == 0) return false;  else return true;  }  void KT\_Gia\_Do\_Thi(int a[][DINH], int n)  {  printf("\nMa Tran Ke:\n");  xuat(a, n);  if(KT\_DTVoHuong(a, n)==1)  {  if(laGiaDoThi(a, n)==true)  {  printf("\nYES");  }  else  {  printf("\nNO");  }  }  else  {  printf("\nNO");  }  }  void DFS(int a[][DINH], int n, int num, bool mask[])  {  if(mask[num] == false)  {  printf("%3d-%c", num, convert(num));  mask[num] = true;  for(int i=1; i <= n; i++)  {  if(a[num][i]!=0)  DFS(a, n, i, mask);  }  }  }  void BFS(int a[][DINH], int n, int num)  {  bool duyeti[20];  int hangdoi[20], phanTuHangDoi, chiSoHienHanh=0;  for(int i=1; i<=n; i++)  {  duyeti[i]=false;  }  phanTuHangDoi = n;  hangdoi[chiSoHienHanh] = num;  chiSoHienHanh++;  while(chiSoHienHanh>0)  {  int x = hangdoi[0];  printf("%3d-%c", x, convert(x));  duyeti[x]=true;  doiMangRaPhiaTruoc(hangdoi, phanTuHangDoi);  chiSoHienHanh--;  for(int i=1; i<=n; i++)  {  if(a[x][i]!=0 && duyeti[i]!=true)  {  hangdoi[chiSoHienHanh] = i;  duyeti[i] = true;  chiSoHienHanh++;  }  }  }  }  void doiMangRaPhiaTruoc(int hangdoi[],int &phanTuHangDoi)  {  for(int i=0; i < phanTuHangDoi-1; i++)  {  hangdoi[i]= hangdoi[i+1];  }  phanTuHangDoi--;  }  void duongDiNganNhat(int a[][DINH], int n, int mangddnn[][DINH], int dinh)  {  int S[DINH], soPhanTu\_S = 1;  //set gia tri ban dau  for(int i=1; i<=n; i++)  {  mangddnn[1][i] = VOCUNG;  mangddnn[2][i] = 0;  mangddnn[3][i] = 0;  }  mangddnn[1][dinh] = 0;    //chua duyet het thi tiep tuc lam  while(soPhanTu\_S <= n)  {    //lay gia tri nho nhat vao S va danh dau  dinh = chiSoCoGiaTriMin(mangddnn, n);  if(dinh == 0)  break;  S[soPhanTu\_S] = dinh;  mangddnn[3][dinh] = DADUYET;  soPhanTu\_S++;      for(int i=1; i<=n; i++)  {  //LAY GIA TRI DUONG DI  int giaTriDuongDi;  if(a[dinh][i]!=0)  {  giaTriDuongDi = a[dinh][i];  }  else  {  giaTriDuongDi = VOCUNG;  }    //KIEM TRA DIEU KIEN  if(mangddnn[1][i] > (mangddnn[1][dinh] + giaTriDuongDi))  {  mangddnn[1][i] = (mangddnn[1][dinh] + giaTriDuongDi);  mangddnn[2][i] = dinh;  }  }  }  } |
| void ddnn\_ThuatToanFloyd(int a[][DINH], int n,int dinh)  {  int A[DINH][DINH], P[DINH][DINH];  for(int i=1; i<=n; i++)  {  for(int j=1; j<=n; j++)  {  if(i==j)  {  A[i][j] = 0;  P[i][j] = 0;  }  else if(a[i][j] == 0)  {  A[i][j] = VOCUNG;  P[i][j] = 0;  }  else  {  A[i][j] = a[i][j];  P[i][j] = j;  }  }  }      printf("\nA0:\n");  xuat(A, n);  printf("\nP0:\n");  xuat(P, n);    for (int k = 1;k <= n;k++)  {  for (int i = 1;i <= n; i++)  for (int j = 1;j <= n; j++)  if(A[i][k] + A[k][j] < A[i][j])  {  A[i][j] = A[i][k] + A[k][j];  P[i][j] = P[i][k];  }  printf("\nA%d:\n", k);  xuat(A, n);  printf("\nP%d:\n", k);  xuat(P, n);  }  printf("\nDuong Di tu dinh %c", convert(dinh));  for(int i=1; i<=n; i++)  {  if(i!=dinh)  {  printf("\n\tDen dinh %c (Do dai bang %3d): ", convert(i), A[dinh][i]);  if((P[dinh][i] == 0)) printf("Khong co duong di");  else{  printf("%3c -> ", convert(dinh));  int tam = dinh;  while(P[tam][i] != i)  {  printf("%3c -> ", convert(P[tam][i]));  tam = P[tam][i];  }  printf("%3c ", convert(i));  }  }  }  }  void chuTrinhEuler(int a[][DINH], int n)  {    if(coChuTrinhEuler(a, n) == true)  {  int copy\_a[DINH][DINH], S[DINH], so\_phan\_tu\_S = 0, Output[DINH \* 2], soPhanTu\_ouput=0, dinh;    do  {  printf("\nNhap dinh: ");  scanf("%d", &dinh);  }while(dinh <=0 || dinh >n);    //copy mang a sang copy\_a  for(int i=1; i<=n; i++)  {  for(int j=1; j<=n; j++)  {  copy\_a[i][j] = a[i][j];  }  }    // dua dinh do vao S  so\_phan\_tu\_S++;  S[so\_phan\_tu\_S] = dinh;  //Tim chu trinh euler  printf("\nChu Trinh Euler la: ");  do{  //lay u la phan tu cuoi cung cua S  int u = S[so\_phan\_tu\_S];    if(conDinh(copy\_a, n, u) == true)  {  //Tim dinh ke va xu ly  for(int i=1; i<=n; i++)  {  if(copy\_a[u][i] != 0)  {  so\_phan\_tu\_S++;  S[so\_phan\_tu\_S] = i;    //danh dau duyet roi  copy\_a[u][i] = 0;  copy\_a[i][u] = 0;  break;  }  }    }  else{  soPhanTu\_ouput++;  Output[soPhanTu\_ouput] = u;  so\_phan\_tu\_S--;  }    }while(so\_phan\_tu\_S > 0);    //xuat duong di  for(int i=soPhanTu\_ouput; i>0; i--)  {  if(i!=1)  printf("%3c ->", convert(Output[i]));  else printf("%3c", convert(Output[i]));  }  }  else if(coDuongDiEuler(a, n)==true)  {  int dem = 0;  for(int i=1; i<=n; i++)  {  dem=0;  for(int j=1; j<=n; j++)  {  if(i==j)  dem += 2 \* a[i][j];  else dem += a[i][j];  }  if(dem % 2 != 0)  {  int copy\_a[DINH][DINH], S[DINH], so\_phan\_tu\_S = 0, Output[DINH \* 2], soPhanTu\_ouput=0, dinh;  dinh = i;  //copy mang a sang copy\_a  for(int i=1; i<=n; i++)  {  for(int j=1; j<=n; j++)  {  copy\_a[i][j] = a[i][j];  }  }    // dua dinh do vao S  so\_phan\_tu\_S++;  S[so\_phan\_tu\_S] = dinh;  //Tim chu trinh euler  printf("\nDuong di Euler la: ");  do{  //lay u la phan tu cuoi cung cua S  int u = S[so\_phan\_tu\_S];    if(conDinh(copy\_a, n, u) == true)  {  //Tim dinh ke va xu ly  for(int i=1; i<=n; i++)  {  if(copy\_a[u][i] != 0)  {  so\_phan\_tu\_S++;  S[so\_phan\_tu\_S] = i;    //danh dau duyet roi  copy\_a[u][i] = 0;  copy\_a[i][u] = 0;  break;  }  }  }  else{  soPhanTu\_ouput++;  Output[soPhanTu\_ouput] = u;  so\_phan\_tu\_S--;  }  }while(so\_phan\_tu\_S > 0);    //xuat duong di  for(int i=soPhanTu\_ouput; i>0; i--)  {  if(i!=1)  printf("%3c ->", convert(Output[i]));  else printf("%3c", convert(Output[i]));  }  }  }  }  else{  printf("\n\t[-Khong ton tai chu trinh va duong di Euler-]\n");  }  } | |
| bool conDinh(int a[][DINH], int n, int dinh)  {  for(int i=1; i<=n; i++)  {  if(a[dinh][i] != 0)  return true;  }  return false;  }  bool coChuTrinhEuler(int a[][DINH], int n)  {  int dem = 0;  if(KT\_DTVoHuong(a, n) == 1)  {  for(int i=1; i<=n; i++)  {  dem=0;  for(int j=1; j<=n; j++)  {  if(i==j)  dem += 2 \* a[i][j];  else dem += a[i][j];  }  if(dem % 2 != 0)  return false;  }  }  else  {  int bacVao=0, bacRa=0;  for(int i=1; i<=n; i++)  {  bacVao=0; bacRa=0;    for(int j=1; j<=n; j++)  {  bacVao += a[j][i];  bacRa += a[i][j];  }    if(bacVao != bacRa)  return false;    }  }  return true;  } | bool coDuongDiEuler(int a[][DINH], int n)  {  int dem = 0, demle=0;  if(KT\_DTVoHuong(a, n) == 1)  {  for(int i=1; i<=n; i++)  {  dem=0;  for(int j=1; j<=n; j++)  {  if(i==j)  dem += 2 \* a[i][j];  else dem += a[i][j];  }  if(dem % 2 != 0)  demle++;  if(demle > 2) return false;  }  }  else  {  int bacVao=0, bacRa=0;  for(int i=1; i<=n; i++)  {  bacVao=0; bacRa=0;    for(int j=1; j<=n; j++)  {  bacVao += a[j][i];  bacRa += a[i][j];  }    if(bacVao != bacRa)  if(bacVao== bacRa-1||bacRa == bacVao-1)  demle++;  else return false;  if(demle>1)  return false;    }  }  return true;  } |