int DT[50][50];

int n;

int Open[50];

int nO = 0;

int Close[50];

int nC = 0;

void DocFile (int DT[][], int &n)

{

FILE \*f = fopen(“…\DothiSearch.txt”, “r”);

fscanf(f, “%d”,&n);

for (int i = 0; i < n; i++)

for (int j = 0; j < n; j++ )

fscanf(f, “%d”,&DT[i][j]);

fclose(f);

}

void AddMember(int x, int L[], int &nL) // Push

{

for (int i = nL; i>= 1; i--)

L[i] = L[i-1];

L[0] = x;

nL++;

}

void AddMember(int x, int L[], int &nL) // EnQueue 🡪 For BFS

{

L[nL] = x;

nL++;

}

int Select(int Open[], int &nO) //Pop

{

int x = Open[0];

for (int i = 0; i<nO-1; i++)

Open[i] = Open[i +1];

nO--;

retrun x;

}

int Search(int x, int L[], int nL) // Trả về 1 nếu x thuộc L, trả về 0 nếu x không thuộc L

{

for (int i = 0; i < nL; i++)

if (L[i] ==x)

return 1;

return 0;

}

void DFS(int S)

{

AddMember(S, Open, nO);

while (nO != 0)

{

int k = Select(Open, nO);

if (Search(k, Close, nC) == 0)

{

Travel(k); // printf(“%d”, k);

AddMember(k, Close, nC);

for(int i = 0; i<n; i ++)

if (DT[k][i] > 0)

if (Search(i, Close, nC) == 0)

AddMember(i, Open, nO);

}

}

}

void main()

{

DocFile(DT, n);

int S;

printf(“Nhập đỉnh bắt đầu: ”);

scanf(“%d”, &S);

DFS(S);

}