**GREEDY**

**Bài 1: Chia tài sản**

Trong ngày sinh nhật, hai anh em Tom và Alice nhận được N(N < 10) đồ chơi. Trên đồ chơi i có ghi giá tiền Xi . Hai anh em quyết định mỗi người phải có trách nhiệm bảo quản một phần số quà và phân chia sao cho chênh lệch tổng giá trị tiền đồ chơi mà mỗi người bảo quản là ít nhất. Hãy giúp Tom và Alice phân chia. Viết cho phép người dùng nhập N đồ vật và giá trị các đồ vật Xi

**#include <iostream>**

**#include <vector>**

**#include <string>**

**using namespace std;**

**void SapXep(vector<int> &taiSan)**

**{**

**for(int i = 0; i < taiSan.size() - 1; i++)**

**{**

**if(taiSan[i] > taiSan[i+1])**

**{**

**int temp = taiSan[i];**

**taiSan[i] = taiSan[i+1];**

**taiSan[i+1] = temp;**

**}**

**}**

**}**

**void In(vector<int> &taiSan)**

**{**

**for(int i:taiSan)**

**{**

**cout << i << " ";**

**}**

**}**

**void ChiaTaiSan(int n, vector<int> &taiSan)**

**{**

**int x;**

**cout << "Nhap gia cua tung do choi: " << endl;**

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

**{**

**cout << "x[" << i << "] = ";**

**cin >> x;**

**taiSan.push\_back(x);**

**}**

**SapXep(taiSan);**

**int sumTom = 0;**

**int sumAlice = 0;**

**vector<int> trai;**

**vector<int> phai;**

**for(int i = n - 1; i >= 0; i--)**

**{**

**if(sumTom <= sumAlice)**

**{**

**trai.push\_back(taiSan[i]);**

**sumTom += taiSan[i];**

**}**

**else**

**{**

**phai.push\_back(taiSan[i]);**

**sumAlice += taiSan[i];**

**}**

**}**

**cout << "Tom nhan duoc do choi co gia tri: ";**

**In(trai);**

**cout << endl;**

**cout << "Alice nhan duoc do choi co gia tri: ";**

**In(phai);**

**}**

**int main()**

**{**

**int n;**

**cout << "Nhap so do choi : ";**

**cin >> n;**

**vector<int> taiSan;**

**ChiaTaiSan(n, taiSan);**

**return 0;**

**}**

**Bài 2: TSP**Có n thành phố, d[i, j] là chi phí để di chuyển từ thành phố i đến thành phố j. (Nếu không có đường đi thì d[i, j] = ∞). Một người muốn đi du lịch qua tất cả các thành phố, mỗi thành phố một lần rồi trở về nơi xuất phát sao cho tổng chi phí là nhỏ nhất. Hãy xác định một đường đi như vậy

**#include <iostream>**

**#include <vector>**

**#include <climits>**

**#define COL 100**

**#define ROW 100**

**using namespace std;**

**void insertMatrix(int n, int arr[ROW][COL])**

**{**

**cout << "Nhap ma tran chi phi: " << endl;**

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

**{**

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

**{**

**cout << "a[" << i << "][" << j << "] = ";**

**cin >> arr[i][j];**

**}**

**}**

**}**

**void TSP(int n, int arr[ROW][COL]) {**

**bool flg[n] = {false};**

**int min = INT\_MAX;**

**int count = 0;**

**cout << "Lo trinh: ";**

**int i = 0;**

**cout << i << " ";**

**flg[i] = true;**

**while(true)**

**{**

**min = INT\_MAX;**

**int next = -1;**

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

**{**

**if(arr[i][j] < min && !flg[j])**

**{**

**min = arr[i][j];**

**next = j;**

**}**

**}**

**if(next == -1) break;**

**cout << next << " ";**

**count += min;**

**i = next;**

**flg[i] = true;**

**}**

**cout << "0";**

**cout << endl;**

**cout << "Tong chi phi thap nhat: " << count + arr[0][n-1] << endl;**

**}**

**int main()**

**{**

**int n;**

**cout << "Nhap so thanh pho: ";**

**cin >> n;**

**int arr[ROW][COL];**

**insertMatrix(n, arr);**

**TSP(n, arr);**

**return 0;**

**}**