Câu 1

#include<stdio.h>

struct hinhtron{

int ma,ox,oy;

float r;

};

int main(){

struct hinhtron c[100];

int n,i;

scanf("%d",&n);

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

scanf("%d %f %d %d",&c[i].ma,&c[i].r,&c[i].ox,&c[i].oy);}

printf("Danh sach hinh tron\n%d",n);

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

printf("\n%d %d %d %.3f",c[i].ma,c[i].ox,c[i].oy,c[i].r);}

}

Câu 2

#include <stdio.h>

struct NhanVien {

char ten[50];

int maNV;

float heSoLuong;

int phuCap;

};

int main() {

struct NhanVien nv[100];

int n,i;

scanf("%d", &n);

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

scanf("%s", nv[i].ten);

scanf("%d", &nv[i].maNV);

scanf("%f", &nv[i].heSoLuong);

scanf("%d", &nv[i].phuCap);

}

printf("Danh sach nhan vien\n");

printf("%d\n", n);

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

printf("%s %d %.2f %d %.2f\n", nv[i].ten, nv[i].maNV, nv[i].heSoLuong, nv[i].phuCap,

nv[i].heSoLuong \* 2000000 + nv[i].phuCap);

}

return 0;

}

Câu 3

#include <stdio.h>

struct SinhVien {

char ten[50];

int maSV;

float Toan,Triet,LTC;

};

int main() {

struct SinhVien sv[100];

int n,i;

scanf("%d", &n);

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

scanf("%s", sv[i].ten);

scanf("%d", &sv[i].maSV);

scanf("%f %f %f", &sv[i].Toan, &sv[i].Triet, &sv[i].LTC);

}

printf("Danh sach sinh vien\n");

printf("%d\n", n);

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

printf("%s %d %.2f %.2f %.2f %.2f\n", sv[i].ten, sv[i].maSV, sv[i].Toan, sv[i].Triet, sv[i].LTC,

(sv[i].Toan+sv[i].Triet+sv[i].LTC)/3);

}

return 0;

}

Câu 4

#include <stdio.h>

struct ThiSinh {

char ten[50];

int SBD;

float Mon1,Mon2,Mon3;

};

int main() {

struct ThiSinh ts[100];

int n,i;

scanf("%d", &n);

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

scanf("%s", ts[i].ten);

scanf("%d", &ts[i].SBD);

scanf("%f %f %f", &ts[i].Mon1, &ts[i].Mon2, &ts[i].Mon3);

}

printf("Danh sach thi sinh\n");

printf("%d\n", n);

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

printf("%s %d %.2f %.2f %.2f %.2f\n", ts[i].ten, ts[i].SBD, ts[i].Mon1, ts[i].Mon2, ts[i].Mon3,

ts[i].Mon1+ts[i].Mon2+ts[i].Mon3);

}

return 0;

}

Câu 5

#include<stdio.h>

struct hinhtron{

int ma,x,y;

float r;

};

void sapxep(struct hinhtron c[],int n){

int i,j;

struct hinhtron temp;

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

for (j =i+1; j < n; j++)

if (c[i].x\*c[i].x+c[i].y\*c[i].y > c[j].x\*c[j].x+c[j].y\*c[j].y) {

temp = c[i];

c[i] = c[j];

c[j] = temp;

}

}

int main(){

struct hinhtron c[100];

int n,i;

scanf("%d",&n);

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

scanf("%d %f %d %d",&c[i].ma,&c[i].r,&c[i].x,&c[i].y);}

sapxep(c,n);

printf("%d %d %d %.3f",c[0].ma,c[0].x,c[0].y,c[0].r);}

Câu 6

#include <stdio.h>

#include <math.h>

struct hinhtron{

int ma;

float r;

int x,y;

};

int main(){

struct hinhtron c[100];

int i,n,xa,ya;

scanf("%d",&n);

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

scanf("%d %f %d %d",&c[i].ma,&c[i].r,&c[i].x,&c[i].y);}

scanf("%d %d",&xa,&ya);

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

float d=sqrt(pow(xa-c[i].x,2)+pow(ya-c[i].y,2));

if(d<c[i].r)

printf("Diem (%d, %d) nam trong hinh tron tam (%d, %d) ban kinh %.3f\n",

xa,ya,c[i].x,c[i].y,c[i].r);

else if(d==c[i].r)

printf("Diem (%d, %d) nam tren hinh tron tam (%d, %d) ban kinh %.3f\n",

xa,ya,c[i].x,c[i].y,c[i].r);

else if(d>c[i].r)

printf("Diem (%d, %d) nam ngoai hinh tron tam (%d, %d) ban kinh %.3f\n",

xa,ya,c[i].x,c[i].y,c[i].r);

}

}

Câu 7

#include<stdio.h>

struct hinhtron{

int ma,ox,oy;

float r;

};

void sapxep(struct hinhtron c[],int n){

int i,j;

struct hinhtron temp;

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

for (j =i+1; j <= n; j++)

if (c[i].r < c[j].r) {

temp = c[i];

c[i] = c[j];

c[j] = temp;

}

}

int main(){

struct hinhtron c[100];

int n,i;

scanf("%d",&n);

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

scanf("%d %f %d %d",&c[i].ma,&c[i].r,&c[i].ox,&c[i].oy);}

printf("Danh sach hinh tron\n%d",n);

sapxep(c,n);

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

printf("\n%d %d %d %.3f",c[i].ma,c[i].ox,c[i].oy,c[i].r);}

}

Câu 8

#include <stdio.h>

struct NhanVien {

char ten[50];

int maNV;

float heSoLuong;

int phuCap;

};

void sapxep(struct NhanVien nv[],int n){

int i,j;

struct NhanVien temp;

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

for (j =i+1; j < n; j++) {

if (nv[i].heSoLuong > nv[j].heSoLuong) {

temp = nv[i];

nv[i] = nv[j];

nv[j] = temp;

}

}

}

}

int main() {

struct NhanVien nv[100];

int n,i;

scanf("%d", &n);

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

scanf("%s", nv[i].ten);

scanf("%d", &nv[i].maNV);

scanf("%f", &nv[i].heSoLuong);

scanf("%d", &nv[i].phuCap);

}

sapxep(nv,n);

printf("Nhan vien co he so luong thap nhat\n");

printf("%s %d %.2f %d %.2f\n", nv[0].ten, nv[0].maNV, nv[0].heSoLuong, nv[0].phuCap,

nv[0].heSoLuong \* 2000000 + nv[0].phuCap);

}

Câu 9

#include <stdio.h>

struct NhanVien {

char ten[50];

int maNV;

float heSoLuong;

int phuCap;

};

void sapxep(struct NhanVien nv[],int n){

int i,j;

struct NhanVien temp;

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

for (j =i+1; j < n; j++) {

if (nv[i].heSoLuong \* 2000000 + nv[i].phuCap<nv[j].heSoLuong \* 2000000 + nv[j].phuCap) {

temp = nv[i];

nv[i] = nv[j];

nv[j] = temp;

}

}

}

}

int main() {

struct NhanVien nv[100];

int n,i;

scanf("%d", &n);

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

scanf("%s", nv[i].ten);

scanf("%d", &nv[i].maNV);

scanf("%f", &nv[i].heSoLuong);

scanf("%d", &nv[i].phuCap);

}

sapxep(nv,n);

printf("Nhan vien co luong lon nhat\n");

printf("%s %d %.2f %d %.2f\n", nv[0].ten, nv[0].maNV, nv[0].heSoLuong, nv[0].phuCap,

nv[0].heSoLuong \* 2000000 + nv[0].phuCap);

}

Câu 10

#include <stdio.h>

struct NhanVien {

char ten[50];

int maNV;

float heSoLuong;

int phuCap;

};

void sapxep(struct NhanVien nv[],int n){

int i,j;

struct NhanVien temp;

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

for (j =i+1; j < n; j++) {

if (nv[i].heSoLuong \* 2000000 + nv[i].phuCap<nv[j].heSoLuong \* 2000000 + nv[j].phuCap) {

temp = nv[i];

nv[i] = nv[j];

nv[j] = temp;

}

}

}

}

int main() {

struct NhanVien nv[100];

int n,i;

scanf("%d", &n);

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

scanf("%s", nv[i].ten);

scanf("%d", &nv[i].maNV);

scanf("%f", &nv[i].heSoLuong);

scanf("%d", &nv[i].phuCap);

}

sapxep(nv,n);

printf("Danh sach nhan vien\n");

printf("%d\n", n);

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

printf("%s %d %.2f %d %.2f\n", nv[i].ten, nv[i].maNV, nv[i].heSoLuong, nv[i].phuCap,

nv[i].heSoLuong \* 2000000 + nv[i].phuCap);

}

return 0;

}

Câu 11

#include <stdio.h>

#include <string.h>

struct NhanVien {

char ten[50];

int maNV;

float heSoLuong;

int phuCap;

};

void sapxep(struct NhanVien nv[],int n){

int i,j;

struct NhanVien temp;

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

for (j =i+1; j < n; j++) {

if (strcmp(nv[i].ten, nv[j].ten)>0) {

temp = nv[i];

nv[i] = nv[j];

nv[j] = temp;

}

}

}

}

int main() {

struct NhanVien nv[100];

int n,i;

scanf("%d", &n);

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

scanf("%s", nv[i].ten);

scanf("%d", &nv[i].maNV);

scanf("%f", &nv[i].heSoLuong);

scanf("%d", &nv[i].phuCap);

}

sapxep(nv,n);

printf("Danh sach nhan vien\n");

printf("%d\n", n);

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

printf("%s %d %.2f %d %.2f\n", nv[i].ten, nv[i].maNV, nv[i].heSoLuong, nv[i].phuCap,

nv[i].heSoLuong \* 2000000 + nv[i].phuCap);

}

return 0;

}

Câu 12

#include<stdio.h>

typedef struct{

char ten[30];

int maSV;

float diemT;

float diemTR;

float diemLTC;}sv;

void nhap(sv a[], int n){

int i;

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

scanf("%s",a[i].ten); scanf("%d",&a[i].maSV);

scanf("%f",&a[i].diemT); scanf("%f",&a[i].diemTR);

scanf("%f",&a[i].diemLTC);}

}

float dtb(sv x){

return ((x.diemT +x.diemTR + x.diemLTC)/3);}

void dtbMax(sv a[], int n){

int i,vt;

sv max=a[0];

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

if(dtb(max)<dtb(a[i])){

max=a[i]; vt=i;}

}

printf("So thu tu cua sinh vien: %d\n",vt);

printf("%s ",max.ten); printf("%d ",max.maSV);

printf("%.2f ",max.diemT); printf("%.2f ",max.diemTR);

printf("%.2f ",max.diemLTC); printf("%.2f ",dtb(max));}

int main(){

sv a[100];

int n;

scanf("%d",&n);

if(n<=100){

nhap(a,n); dtbMax(a,n);}

}

Câu 13

#include<stdio.h>

#include<string.h>

typedef struct {

char ten[30];

int maSV;

float diemT;

float diemTR;

float diemLTC; }sv;

void nhap(sv a[], int n){

int i;

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

scanf("%s",a[i].ten);

scanf("%d",&a[i].maSV);

scanf("%f",&a[i].diemT);

scanf("%f",&a[i].diemTR);

scanf("%f",&a[i].diemLTC);}

}

float dtb(sv x){

return ((x.diemT+x.diemTR+x.diemLTC)/3);}

void dtbMin(sv a[], int n){

int i,vt=0;

sv min=a[0];

for(i=1;i<n;i++){

if(dtb(min)>=dtb(a[i])){

min=a[i]; vt=i;}

}

printf("So thu tu cua sv: %d\n",vt);

printf("%s ",min.ten); printf("%d ",min.maSV);

printf("%.2f ",min.diemT); printf("%.2f ",min.diemTR);

printf("%.2f ",min.diemLTC); printf("%.2f ",dtb(min));

}

int main(){

sv a[100];

int n;

scanf("%d",&n);

nhap(a,n);

dtbMin(a,n);

}

Câu 14

#include <stdio.h>

struct SinhVien {

char ten[50];

int maSV;

float Toan,Triet,LTC;

};

int main() {

struct SinhVien sv[100];

int n,i,dem=0;

scanf("%d", &n);

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

scanf("%s", sv[i].ten);

scanf("%d", &sv[i].maSV);

scanf("%f %f %f", &sv[i].Toan, &sv[i].Triet, &sv[i].LTC);

}

printf("Danh sach sinh vien dat hoc bong\n");

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

if(sv[i].Toan>=5.5 && sv[i].Triet>=5.5 && sv[i].LTC>=5.5 && (sv[i].Toan+sv[i].Triet+sv[i].LTC)/3>=7){

printf("%s %d %.2f %.2f %.2f %.2f\n", sv[i].ten, sv[i].maSV, sv[i].Toan, sv[i].Triet, sv[i].LTC,

(sv[i].Toan+sv[i].Triet+sv[i].LTC)/3);

dem++;}

}

printf("So sinh vien dat hoc bong: %d",dem);

return 0;

}

Câu 15

#include <stdio.h>

struct SinhVien {

char ten[50];

int maSV;

float Toan,Triet,LTC;

};

int main() {

struct SinhVien sv[100];

int n,i,dem=0;

scanf("%d", &n);

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

scanf("%s", sv[i].ten);

scanf("%d", &sv[i].maSV);

scanf("%f %f %f", &sv[i].Toan, &sv[i].Triet, &sv[i].LTC);

}

printf("Danh sach sinh vien hoc lai\n");

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

if(sv[i].Toan<4 || sv[i].Triet<4 || sv[i].LTC<4){

printf("%s %d %.2f %.2f %.2f %.2f\n", sv[i].ten, sv[i].maSV, sv[i].Toan, sv[i].Triet, sv[i].LTC,

(sv[i].Toan+sv[i].Triet+sv[i].LTC)/3);

dem++;}

}

printf("So sinh vien phai hoc lai: %d",dem);

return 0;

}

Câu 16

#include <stdio.h>

struct SinhVien {

char ten[50];

int maSV;

float Toan,Triet,LTC;

};

int main() {

struct SinhVien sv[100];

int n,i,dem=0;

scanf("%d", &n);

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

scanf("%s", sv[i].ten);

scanf("%d", &sv[i].maSV);

scanf("%f %f %f", &sv[i].Toan, &sv[i].Triet, &sv[i].LTC);

}

printf("Danh sach sinh vien hoc lai\n");

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

if((sv[i].Toan<4 && sv[i].Triet<4)|| (sv[i].Triet<4 && sv[i].LTC<4) || (sv[i].Toan<4 && sv[i].LTC<4)){

printf("%s %d %.2f %.2f %.2f %.2f\n", sv[i].ten, sv[i].maSV, sv[i].Toan, sv[i].Triet, sv[i].LTC,

(sv[i].Toan+sv[i].Triet+sv[i].LTC)/3);

dem++;}

}

printf("So sv phai hoc lai: %d",dem);

return 0;

}

Câu 17

#include <stdio.h>

struct SinhVien {

char ten[50];

int maSV;

float Toan,Triet,LTC;

};

void sapxep(struct SinhVien sv[],int n){

int i,j;

struct SinhVien temp;

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

for (j =i+1; j < n; j++) {

if ((sv[i].Toan+sv[i].Triet+sv[i].LTC)/3 < (sv[j].Toan+sv[j].Triet+sv[j].LTC)/3) {

temp = sv[i];

sv[i] = sv[j];

sv[j] = temp;

}

}

}

}

int main() {

struct SinhVien sv[100];

int n,i;

scanf("%d", &n);

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

scanf("%s", sv[i].ten);

scanf("%d", &sv[i].maSV);

scanf("%f %f %f", &sv[i].Toan, &sv[i].Triet, &sv[i].LTC);

}

sapxep(sv,n);

printf("Danh sach sinh vien\n");

printf("%d\n", n);

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

printf("%s %d %.2f %.2f %.2f %.2f\n", sv[i].ten, sv[i].maSV, sv[i].Toan, sv[i].Triet, sv[i].LTC,

(sv[i].Toan+sv[i].Triet+sv[i].LTC)/3);

}

return 0;

}

Câu 18

#include <stdio.h>

struct SinhVien {

char ten[50];

int maSV;

float Toan,Triet,LTC;

};

void sapxep(struct SinhVien sv[],int n){

int i,j;

struct SinhVien temp;

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

for (j =i+1; j < n; j++) {

if (sv[i].maSV > sv[j].maSV) {

temp = sv[i];

sv[i] = sv[j];

sv[j] = temp;

}

}

}

}

int main() {

struct SinhVien sv[100];

int n,i;

scanf("%d", &n);

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

scanf("%s", sv[i].ten);

scanf("%d", &sv[i].maSV);

scanf("%f %f %f", &sv[i].Toan, &sv[i].Triet, &sv[i].LTC);

}

sapxep(sv,n);

printf("Danh sach sinh vien\n");

printf("%d\n", n);

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

printf("%s %d %.2f %.2f %.2f %.2f\n", sv[i].ten, sv[i].maSV, sv[i].Toan, sv[i].Triet, sv[i].LTC,

(sv[i].Toan+sv[i].Triet+sv[i].LTC)/3);

}

return 0;

}

Câu 19

#include <stdio.h>

#include <string.h>

struct SinhVien {

char ten[50];

int maSV;

float Toan,Triet,LTC;

};

void sapxep(struct SinhVien sv[],int n){

int i,j;

struct SinhVien temp;

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

for (j =i+1; j < n; j++) {

if (strcmp(sv[i].ten, sv[j].ten)>0) {

temp = sv[i];

sv[i] = sv[j];

sv[j] = temp;

}

}

}

}

int main() {

struct SinhVien sv[100];

int n,i;

scanf("%d", &n);

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

scanf("%s", sv[i].ten);

scanf("%d", &sv[i].maSV);

scanf("%f %f %f", &sv[i].Toan, &sv[i].Triet, &sv[i].LTC);

}

sapxep(sv,n);

printf("Danh sach sinh vien\n");

printf("%d\n", n);

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

printf("%s %d %.2f %.2f %.2f %.2f\n", sv[i].ten, sv[i].maSV, sv[i].Toan, sv[i].Triet, sv[i].LTC,

(sv[i].Toan+sv[i].Triet+sv[i].LTC)/3);

}

return 0;

}

Câu 20

#include <stdio.h>

struct ThiSinh {

char ten[50];

int SBD;

float Mon1,Mon2,Mon3;

};

int main() {

struct ThiSinh ts[100];

int n,i,dem=0;

scanf("%d", &n);

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

scanf("%s", ts[i].ten);

scanf("%d", &ts[i].SBD);

scanf("%f %f %f", &ts[i].Mon1, &ts[i].Mon2, &ts[i].Mon3);

}

printf("Danh sach thi sinh thi do\n");

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

if(ts[i].Mon1>=1 && ts[i].Mon2>=1 && ts[i].Mon3>=1 && ts[i].Mon1+ts[i].Mon2+ts[i].Mon3>=15){

printf("%s %d %.2f %.2f %.2f %.2f\n", ts[i].ten, ts[i].SBD, ts[i].Mon1, ts[i].Mon2, ts[i].Mon3,

ts[i].Mon1+ts[i].Mon2+ts[i].Mon3);

dem++;}

}

printf("So thi sinh thi do: %d",dem);

return 0;

}

Câu 21

Viết chương trình nhập vào một danh sách gồm n thí sinh thi đại học (n<=100). In danh sách các thí sinh thi trượt ra màn hình. Biết thông tin của một thí sinh gồm: tên, số báo danh, điểm môn 1, điểm môn 2, điểm môn 3. Thí sinh thi trượt khi tổng điểm < 15.00 hoặc có một môn mà  điểm <1.00.

Input:

+ Dòng thứ nhất nhập vào n là số thí sinh

+ Dòng tiếp theo nhập vào thông tin của các thí sinh theo thứ tự tên, số báo danh, điểm môn 1, điểm môn 2, điểm môn 3.

Output:

+ Dòng thứ nhất in ra thông báo "Danh sach thi sinh thi truot"

+ Các dòng tiếp theo, mỗi dòng in ra thông tin của từng thí sinh. Các thông tin cách nhau dấu cách (thứ tự: tên, số báo danh, điểm môn 1, điểm môn 2, điểm môn 3), tổng điểm in ở cuối mỗi dòng.

+ Dòng cuối in ra thông báo "So thi sinh thi truot: X". X là số lượng thí sinh thi trượt.

Constrains: 1<=n<=100, điểm các môn và tổng điểm có độ chính xác 2 chữ số thập phân, số báo danh là số nguyên dương, tên thí sinh không chứa dấu cách.

**For example:**

| **Input** | **Result** |
| --- | --- |
| 5  Huy 87 5.00 3.00 3.00  Lan 2 3.00 2.00 3.00  Anh 25 7.00 9.00 10.00  Van 35 0.00 10.00 9.00  Trang 76 9.00 9.00 9.00 | Danh sach thi sinh thi truot  Huy 87 5.00 3.00 3.00 11.00  Lan 2 3.00 2.00 3.00 8.00  Van 35 0.00 10.00 9.00 19.00  So thi sinh thi truot: 3 |

#include <stdio.h>

struct ThiSinh {

char ten[50];

int SBD;

float Mon1,Mon2,Mon3;

};

int main() {

struct ThiSinh ts[100];

int n,i,dem=0;

scanf("%d", &n);

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

scanf("%s", ts[i].ten);

scanf("%d", &ts[i].SBD);

scanf("%f %f %f", &ts[i].Mon1, &ts[i].Mon2, &ts[i].Mon3);

}

printf("Danh sach thi sinh thi truot\n");

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

if(ts[i].Mon1<1 || ts[i].Mon2<1 || ts[i].Mon3<1 || ts[i].Mon1+ts[i].Mon2+ts[i].Mon3<15){

printf("%s %d %.2f %.2f %.2f %.2f\n", ts[i].ten, ts[i].SBD, ts[i].Mon1, ts[i].Mon2, ts[i].Mon3,

ts[i].Mon1+ts[i].Mon2+ts[i].Mon3);

dem++;}

}

printf("So thi sinh thi truot: %d",dem);

return 0;

}

Câu 22

Viết chương trình nhập vào một danh sách gồm n thí sinh thi đại học (n<=100). In thông tin thí sinh thủ khoa (thi đỗ và có tổng điểm cao nhất) ra màn hình.

   +Nếu có nhiều thủ khoa thì in ra thủ khoa có số thứ tự nhỏ nhất trong danh sách sách.

   + Nếu không có thủ khoa (mọi thí sinh đều trượt) thì thông báo "Khong co ai thi do"

   + Biết thông tin của một thí sinh gồm: tên, số báo danh, điểm môn 1, điểm môn 2, điểm môn 3.

   + Thí sinh thi đỗ khi tổng điểm >= 15.00 và điểm các môn >=1.00.

Input:

+ Dòng thứ nhất nhập vào n là số thí sinh

+ Dòng tiếp theo nhập vào thông tin của các thí sinh theo thứ tự tên, số báo danh, điểm môn 1, điểm môn 2, điểm môn 3.

Output:

Nếu tìm được thủ khoa:

+ Dòng thứ nhất in ra thông báo "So thu tu cua thu khoa: X", X là số thự tự của thí sinh đỗ thủ khoa tìm được theo yêu cầu.

+ Dòng thứ hai in ra thông tin của thí sinh thủ khoa tìm được. Các thông tin cách nhau dấu cách (thứ tự: tên, số báo danh, điểm môn 1, điểm môn 2, điểm môn 3), tổng điểm in ở cuối mỗi dòng.

Nếu KHÔNG tìm được thủ khoa:

+ In ra thông báo "Khong co ai thi do"

Constrains: 1<=n<=100, điểm các môn và tổng điểm có độ chính xác 2 chữ số thập phân, số báo danh là số nguyên dương, tên thí sinh không chứa dấu cách.

**For example:**

| **Input** | **Result** |
| --- | --- |
| 5  Huy 87 5.00 3.00 3.00  Trang 76 9.00 9.00 9.00  Lan 2 3.00 2.00 3.00  Anh 25 8.00 9.00 10.00  Van 35 0.00 10.00 9.00 | So thu tu cua thu khoa: 1  Trang 76 9.00 9.00 9.00 27.00 |

#include<stdio.h>

struct SV{

char ten[30];

int ma;

float dt,dtr,dltc,dtb;

};

int main(){

SV a[100];

int i,n,d=0;

scanf("%d", &n);

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

scanf("%s%d%f%f%f",a[i].ten, &a[i].ma,&a[i].dt,&a[i].dtr,&a[i].dltc);

a[i].dtb=(a[i].dt+a[i].dtr+a[i].dltc);

}

float max=-1e9-1;

int vt;

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

if(a[i].dt>=1 && a[i].dtr>=1 && a[i].dltc>=1 && a[i].dtb>=15 && a[i].dtb>max){

max=a[i].dtb;

vt=i;

d++;}

}

if(d==0) {

printf("Khong co ai thi do");}

else printf("So thu tu cua thu khoa: %d\n%s %d %.2f %.2f %.2f %.2f\n",vt,a[vt].ten, a[vt].ma,a[vt].dt,a[vt].dtr,a[vt].dltc, a[vt].dtb);

return 0;

}

Câu 23

Một câu được gọi là đối xứng nếu sau khi

* loại bỏ các dấu cách, dấu câu (chấm, phẩy, hỏi, chấm than, ..) và các ký tự đặc biệt (@, #, $, ..), chỉ giữ lại chữ cái, chữ số và
* chuyển các chữ hoa thành chữ thường

thì nhận được một câu mới đọc theo thứ tự từ trái sang phải giống như đọc theo thứ tự từ phải sang trái. Ví dụ câu sau là một câu đối xứng:

Now I see bees, I won!Now I see bees, I won!

Hãy viết chương trình nhập vào một câu s� và kiểm tra xem s� có đối xứng không?

+ Đầu vào: Câu s�

+ Đầu ra: in ra TrueTrue nếu s� đối xứng, ngược lại in ra FalseFalse

+ Ràng buộc: số kí tự trong s� không vượt quá 204

!! Chú ý không dùng hàm strlwrstrlwr vì đó không phải là một "standard built-in function".

**For example:**

| **Input** | **Result** |
| --- | --- |
| Borrow or rob? | True |
| I tried my best to pass the exam! | False |

#include<stdio.h>

#include<string.h>

#include<ctype.h>

int main(){

char arr[204],arr1[204];

fgets(arr,1000,stdin);

int len,k=0,flag;

len=strlen(arr);

for (int i=0;i<len;i++){

if(isalnum(arr[i])){

arr1[k]=tolower(arr[i]);

k++;

}

}

arr1[k]='\0';

flag=1;

for(int i=0,j=k-1;i<j;i++,j--){

if(arr1[i]!=arr1[j]){

flag=0;

break;

}

}

if(flag==0){

printf("False");

}

else{

printf("True");

}

return 0;

}

Giải thích rõ

#include<stdio.h>

#include<string.h>

#include<ctype.h>

int main(){

char arr[204],arr1[204];

fgets(arr,204,stdin);//Lấy dữ liệu nhập từ bàn phím vào chuỗi arr

int len,k=0,flag;

len=strlen(arr); //Hàm lấy độ dài của chuỗi

for (int i=0;i<len;i++){

if(isalnum(arr[i])) //Nếu ký tự kiểm tra là chữ cái hoặc số

{

arr1[k]=tolower(arr[i]); //Đưa ký tự về dạng chữ thường

k++;

}

}

arr1[k]='\0'; //Để ký tự Null/rỗng vào cuối chuỗi arr1

flag=1; //Mặc định chuỗi là chuỗi đối xứng

for(int i=0,j=k-1;i<j;i++,j--) //Cho i chạy từ đầu chuỗi, j chạy từ cuối chuỗi

{

if(arr1[i]!=arr1[j])

{

flag=0; //Chuỗi không phải đối xứng

break;

}

}

if(flag==0) //Nếu chuỗi không phải đối xứng

{

printf("False");

}

else //<=> (flag == 1) Chuỗi là đối xứng

{

printf("True");

}

return 0;

}

Câu 24

Viết chương trình nhập vào hai xâu kí tự s,t�,� và in ra số lần xuất hiện của xâu t� trong s�

+ Đầu vào: xâu s� trên dòng thứ nhất, xâu t� trên dòng thứ hai

+ Đầu ra: Số lần xuất hiện của xâu t� trong xâu s�

+ Ràng buộc: Số kí tự trong mỗi xâu s,t�,� không quá 30

**For example:**

| **Input** | **Result** |
| --- | --- |
| this is just a sample test  is | 2 |

#include <stdio.h>

#include <string.h>

int find(char\* s, char\* t)

{

int count = 0;

char\* pos = strstr(s, t); //strstr: hàm trả về vị trí xuất hiện của xâu t trong xâu s

//nếu xâu t không xuất hiện trong xâu s thì hàm này trả về NULL

while (pos != NULL)

{

count++;

pos = strstr(pos + strlen(t), t); //tìm kiếm vị trí xuất hiện của xâu t trong xâu s(tính từ vị trí xuất hiện xâu t cuối cùng).

}

return count;

}

int main()

{

char s[100];

char t[100];

fgets(s, sizeof(s), stdin);

fgets(t, sizeof(t), stdin);

s[strlen(s) - 1] = t[strlen(t) - 1] = '\0'; //Bỏ đi ký tự enter cuối cùng

if((strlen(s) <= 30) && (strlen(t) <= 30))

printf("%d", find(s, t));

}Câu 27

Giả sử mật khẩu để đăng nhập vào tài khoản của bạn là "123456a@B". Bạn hãy viết chương trình cho yêu cầu nhập vào mật khẩu. Nếu nhập sai (không quá 5 lần) thì cho nhập lại. Nếu nhập sai quá 5 lần thì thông báo "Tai khoan bi khoa, X". Nếu nhập đúng thì thông báo "Dang nhap thanh cong, X".

Trong đó: X là số lần bạn nhập mật khẩu.

Input:

+ một chuỗi ký tự không có dấu cách

Output:

+ Thông báo như đề bài

Constrains:

+ Mật khẩu là chuỗi ký tự không chứa dấu cách, dấu tab, độ dài tối đa là 30.

**For example:**

| **Input** | **Result** |
| --- | --- |
| p1KaLmIj2k  B7fZp5Tm1q  bi6dss92C3  1aPGJK9Lvk  7DMPDONXrw | Tai khoan bi khoa, 5 |

#include<stdio.h>

#include<string.h>

int main(){

char mk[30]="123456a@B";

char s[30];

int temp=0, d=0;

while(temp<5){

temp++;

scanf("%s",s);

if(strcmp(s,mk)==0){

printf("Dang nhap thanh cong, %d ",temp);

d++;

break;}

}

if(temp==5&&d==0)

printf("Tai khoan bi khoa, 5");

}

Câu 29

Viết chương trình nhập vào một chuỗi ký tự. In ra số lần xuất hiện từng ký tự trong chuỗi.

Input: Chuỗi ký tự không bao gồm các ký tự điều khiển.

Output: Gồm nhiều dòng, mỗi dòng in ra X: Y

Trong đó: X là ký tự, Y là số lần xuất hiện của X trong chuỗi ban đầu. Thứ tự in ra tuân theo mã ASCII của ký tự.

Constrains: Chuỗi ký tự không bao gồm các ký tự điều khiển, có độ dài không quá 100.

Ví dụ:

Input: Chao Cac Ban

Output:

 : 2 //dấu cách xuất hiện 2 lần

B: 1

C: 2

a: 3

c: 1

h: 1

n: 1

o: 1

**For example:**

| **Input** | **Result** |
| --- | --- |
| Chao Cac Ban | : 2  B: 1  C: 2  a: 3  c: 1  h: 1  n: 1  o: 1 |

#include<stdio.h>

#include<string.h>

#include<ctype.h>

int main(){

int X[256];

char s[100];

fflush(stdin);

scanf("%[^\n]", s);

for(int i=0; i<256; i++){

X[i]=0;}

for(unsigned int i=0; i<strlen(s); i++){

X[(int)s[i]]++;}

for(unsigned int i=0; i<256; i++){

if(X[i]>0){

printf("%c: %d\n", i, X[i]);

}

}

}

Câu 30

Viết chương trình nhập vào một chuỗi gồm các chữ cái và dấu cách. Chuẩn hóa chuỗi vừa nhập thành dạng tên riêng. Chuỗi ký tự ở dạng tên riêng là chuỗi gồm các từ thoả mãn đồng thời các điều kiện sau:

1) Các từ cách nhau đúng một dấu cách.

2) Đầu và cuối chuỗi không có dấu cách.

3) Ký tự đầu mỗi từ là chữ hoa.

4) Các ký tự còn lại là chữ thường.

Ví dụ: Nguyen Van Hung là chuỗi ở dạng tên riêng, còn NGuyen  VAn hUng thì không.

Input: chuỗi ký tự bao gồm chữ cái, dấu cách.

Output: chuỗi ký tự ở dạng tên riêng.

Constrains: chuỗi ký tự chỉ chứa các chữ cái, dấu cách, độ dài không quá 255.

**For example:**

| **Input** | **Result** |
| --- | --- |
| NGuyen VAn AnH | Nguyen Van Anh |

#include<stdio.h>

#include<string.h>

#include<ctype.h>

void Xoa(char s[], unsigned int k, unsigned int l){

for(unsigned int i=k; i<l-1; i++){

s[i]=s[i+1];}

s[l-1]='\0';}

void XoaCach(char s[], unsigned int l){

for(unsigned int i=0; i<l; i++){

if(s[i]==' ' && s[i+1]==' '){

Xoa(s,i+1,l);

i--; l--;}

}

if(s[0]==' ')

Xoa(s,0,l);

l=strlen(s);

if(s[l-1]==' ')

s[l-1]='\0';

}

void Hoa(char s[], unsigned int l){

s[0]=toupper(s[0]);

for(unsigned int i=1; i<l; i++){

if(isalpha(s[i])==1 && s[i-1]!=' ')

s[i]=tolower(s[i]);

if(s[i]==' ')

s[i+1]=toupper(s[i+1]);}

}

int main(){

char s[100];

fflush(stdin);

scanf("%[^\n]", s);

unsigned int l=strlen(s);

XoaCach(s,l);

l=strlen(s);

Hoa(s,l);

printf("%s", s);

}

Câu 31

Viết chương trình nhập vào một chuỗi ký tự chỉ gồm chữ cái, chữ số. Nếu các chữ cái, chữ số trong chuỗi xuất hiện nhiều hơn một lần thì xóa ở lần xuất hiện thứ 2, 3, ... In chuỗi kết quả ra màn hình.

Input: chuỗi ký tự có độ dài <=100.

Output: in ra chuỗi ký tự sao cho mỗi ký tự chỉ xuất hiện không quá một lần.

Constrains: chuỗi ký tự chỉ chứa chữ cái, chữ số, độ dài không quá 100.

**For example:**

| **Input** | **Result** |
| --- | --- |
| Chaocacbansinhvien | Chaocbnsive |

#include<stdio.h>

#include<string.h>

#include<ctype.h>

int main(){

int X[256];

char s[100];

unsigned int i;

fflush(stdin);

scanf("%s", s);

for(i=0; i<256; i++){

X[i]=0;}

for(i=0; i<strlen(s); i++){

X[(int)s[i]]++;

if(X[(int)s[i]]<2)

printf("%c",s[i]);

}

}

Câu 25

Viết chương trình đếm số từ trong một xâu kí tự (giả sử các từ cách nhau bởi dấu cách)

+ Đầu vào: Xâu kí tự s�

+ Đầu ra: Số từ trong xâu s�

+ Ràng buộc: Số kí tự trong xâu s� nhỏ hơn 100

**For example:**

| **Input** | **Result** |
| --- | --- |
| Oh how I wish to go down with the sun | 10 |

#include<stdio.h>

#include<string.h>

int main(){

char s[100];

fgets(s, sizeof(s), stdin);

int length=strlen(s) - 1;

int count = (s[0] != ' ');

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

{

if (s[i] == ' ' && s[i + 1] != ' ')

{

count++;

}

}

printf("%d",count);

}

Câu 26 partical correct

Ở Việt Nam, ngày tháng hợp lệ là ngày tháng tuân theo một trong bốn định dạng sau:

1. dd/mm/yyyy
2. d/mm/yyyy
3. dd/m/yyyy
4. d/m/yyyy

trong đó d, m, yd, m, y là các chữ số.

Viết chương trình nhập vào một xâu s và kiểm tra xem s có tuân theo định dạng ngày/tháng /năm không?

Đầu vào: Xâu kí tự s

Ràng buộc: s chứa không quá 20 kí tự

Đầu ra: In ra ValidValid nếu s tuân theo định dạng ngày tháng năm, ngược lại in ra Not validNot valid

**For example:**

| **Input** | **Result** |
| --- | --- |
| 01/01/2011 | Valid |
| 1/01/2011 | Valid |
| 01/1/2011 | Valid |
| 1/1/2011 | Valid |
| 11/11/201 | Not valid |

#include <stdio.h>

#include <string.h>

int validdm(char\* a, int\* ii)

{

int i = 0;

while(a[i] != '/')

{

if(a[i] <= '9' && a[i] >= '0')

{

i++;

if(i > 2)

{

return 0;

}

}

else

{

return 0;

}

}

if(a[0] == '/')

return 0;

\*ii = i;

return 1;

}

int validy(char\* a)

{

int i = 0;

while(a[i] <= '9' && a[i] >= '0')

{

if(i > 3)

{

return 0;

}

i++;

}

if(i < 4)

{

return 0;

}

return 1;

}

int main()

{

char a[1000];

int index = 0;

int index2 = 0;

fgets(a, sizeof(a), stdin);

a[strlen(a) - 1] = '\0';

if(strlen(a) <= 20)

{

if(validdm(a, &index))

if(validdm(a + index + 1, &index2))

if(validy(a + index + index2 + 1 + 1))

{

printf("Valid");

return 0;

}

printf("Not valid");

}

}

À lan ơi t bảo này. T cũng không chắc m có người yêu hay chưa. Nhưng mà t thích m kể từ lần đi chùa đó chắc m đã biết rồi. Mà tất nhiên t cũng không thể đòi hỏi 1 mối quan hệ từ m được. nếu m đồng ý thì tương lai sẽ có nhiều điều hay ho ấy mà nếu không được thì m có thể giả điếc coi như t chưa nói j nhé!