2001210004 – Lưu Đức Vinh

Bài về nhà Buổi 1:

Bài 6:

#define \_CRT\_NONSTDC\_NO\_WARNINGS

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <conio.h>

#include <string.h>

void Tinh\_Thoigian\_Nghihuu (char \*x, char \*s, int &d, int &m, int &y);

int main ()

{

char x[30], s[10];

int d, m, y;

try

{

Tinh\_Thoigian\_Nghihuu(x, s, d, m, y);

}

catch (const char \*msg)

{

printf("%s\n", msg);

}

catch (int Num)

{

printf("Loi ma: %d\n", Num);

}

return 0;

getch ();

}

void Tinh\_Thoigian\_Nghihuu (char \*x, char \*s, int &d, int &m, int &y)

{

printf("Nhap ho ten: ");

fflush(stdin);

gets(x);

do

{

printf("\t\t- Nhap ngay: ");

scanf\_s("%d", &d);

}

while (d <= 0 || d > 31);

do

{

printf("\t\t- Nhap thang: ");

scanf\_s("%d", &m);

}

while (m <= 0 || m > 12);

printf("\t\t- Nhap vao nam: ");

scanf\_s("%d", &y);

printf("Nhap gioi tinh: ");

fflush(stdin);

gets(s);

if ((2022 - y) < 18)

{

throw 101;

}

else if (stricmp(s, "Nam") != 0 && stricmp(s, "Nu") != 0)

{

throw "errorcode";

}

else

{

printf("Ho ten: %s\n", x);

printf("Gioi tinh: %s\n", s);

printf("Sinh ngay: %d/%d/%d\n", d, m, y);

printf("Hien tai (nam 2022) %s da duoc %d tuoi\n", x, (2022 - y));

if (stricmp(s, "Nam") == 0)

{

printf("Thoi gian %s duoc nghi huu la thang %d/%d\n", x, m, y + 62);

}

else

{

printf("Thoi gian %s duoc nghi huu la thang %d/%d\n", x, m, y + 60);

}

}

}

Bài 7 và Bài 11:

#define \_CRT\_NONSTDC\_NO\_WARNINGS

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <conio.h>

#include <stdlib.h>

#include <math.h>

void NhapM1C (int \*&a, int &n);

void nhapM1C\_Random (int \* &a, int &n);

void xuatM1C(int \*a, int n);

void Max\_M1C (int \*a, int n);

int \*PT\_Chan\_Max (int \*a, int n);

int \*PT\_Le\_Min (int \*a, int n);

void XoaVT (int \*a, int &n, int x);

void XoatPT\_0 (int \*a, int &n);

void ThemPT\_SauPTFirst (int \*a, int &n, int &x);

int kiemtrascp (int n);

int TongSCP (int \*a, int n);

void XuatPT\_CucDai (int \*a, int n);

void XuatPT\_CucTieu (int \*a, int n);

void themPT (int \*a, int &n, int &x, int &vitri);

void XoaPT (int \*a, int &n, int &x);

int kiemtrachanle (int \*a, int n);

void swap (int &x, int &y);

void DuaChanLenDau\_LeXuongDuoi (int \*a, int n);

int main ()

{

int \*a;

int n, chon;

do

{

printf("1.Nhap M1C\n");

printf("2.Nhap M1C random\n");

printf("3.Xuat M1C\n");

printf("4.PT Max trong mang\n");

printf("5.Xuat dia chi PT chan lon nhat va PT le nho nhat\n");

printf("6.Xoa PT co gia tri 0\n");

printf("7.Them PT x vao sau PT dau tien\n");

printf("8.Tinh tong cac PT la so chinh phuong\n");

printf("9.Xuat cac so cuc dai trong mang\n");

printf("10.Xuat cac so cuc tieu trong mang\n");

printf("11.Xoa pt tai vi tri k\n");

printf("12.Them PT x tai vi tri k\n");

printf("13.Chuyen so chan len dau mang va so le xuong cuoi mang\n");

printf("14.Kiem tra mang co chua chan le xen ke khong\n");

printf("Chon chuc nang: "); scanf\_s("%d", &chon);

switch (chon)

{

case 1:

{

NhapM1C(a, n);

break;

}

case 2:

{

nhapM1C\_Random(a, n);

break;

}

case 3:

{

xuatM1C(a, n);

break;

}

case 4:

{

Max\_M1C(a, n);

break;

}

case 5:

{

int \*b[1];

try

{

b[0] = PT\_Chan\_Max(a, n);

printf("\nPT chan Max co gia tri %d va dia chi o nho la %x\n", \*b[0], b[0]);

}

catch (const char \*msg)

{

printf("\n%s\n", msg);

}

try

{

b[1] = PT\_Le\_Min(a, n);

printf("\nPT le Min co gia tri %d va dia chi o nho la %x\n", \*b[1], b[1]);

}

catch (const char \*msg)

{

printf("\n%s\n", msg);

}

}

case 6:

{

XoatPT\_0(a, n);

break;

}

case 7:

{

int x;

ThemPT\_SauPTFirst(a, n, x);

break;

}

case 8:

{

printf("Tong cac PT la so chinh phuong la: %d\n", TongSCP(a, n));

break;

}

case 9:

{

XuatPT\_CucDai(a, n);

break;

}

case 10:

{

XuatPT\_CucTieu(a, n);

break;

}

case 11:

{

int x;

XoaPT(a, n, x);

break;

}

case 12:

{

int x, vitri;

themPT(a, n, x, vitri);

break;

}

case 13:

{

DuaChanLenDau\_LeXuongDuoi(a, n);

break;

}

case 14:

{

int i = kiemtrachanle(a, n);

if (i != 0)

printf("\nMang co chua chan le xen ke\n");

else

printf("\nMang khong chua chan le xen ke\n");

break;

}

}

}

while (chon != 0);;

return 0;

getch ();

}

void NhapM1C (int \*&a, int &n)

{

printf("Nhap so luong PT mang: "); scanf\_s("%d", &n);

a = (int \*)malloc(n\*sizeof(int));

srand((unsigned)time(NULL));

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

{

printf("Nhap PT a[%d]: ", i); scanf\_s("%d", &a[i]);

}

}

void nhapM1C\_Random (int \*&a, int &n)

{

printf("Nhap so luong PT mang: "); scanf("%d", &n);

a = (int \*)malloc(n\*sizeof(int));

srand((unsigned)time(NULL));

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

{

\*(a + i) = rand() % 100;

}

}

void xuatM1C(int \*a, int n)

{

printf("Xuat mang:\n");

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

{

printf("%5d", a[i]);

}

printf("\n");

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

{

printf("\nPhan tu thu %d co gia tri %d va dia chi o nho la %x\n", i, \*(a + i), a + i);

}

}

void Max\_M1C (int \*a, int n)

{

int \*Max = a;

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

{

if (\*Max < a[i])

{

\*Max = a[i];

}

}

printf("PT Max cua mang co gia tri %d va dia chi o nho la %x\n", \*Max, Max);

}

int \*PT\_Chan\_Max (int \*a, int n)

{

int i, d, vt;

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

{

if (a[i] % 2 == 0)

{

d = i;

break;

}

}

if (i == n)

{

throw "Khong co PT chan trong mang";

}

vt = d;

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

{

if (a[i] % 2 == 0 && a[i] > a[vt])

{

vt = i;

}

}

return &a[vt];

}

int \*PT\_Le\_Min (int \*a, int n)

{

int i, d, vt;

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

{

if (a[i] % 2 != 0)

{

d = i;

break;

}

}

if (i == n)

{

throw "Khong co PT le trong mang";

}

vt = d;

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

{

if (a[i] % 2 != 0 && a[i] < a[vt])

{

vt = i;

}

}

return &a[vt];

}

void XoaVT (int \*a, int &n, int x)

{

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

{

a[i] = a[i + 1];

}

n--;

}

void XoatPT\_0 (int \*a, int &n)

{

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

{

if (a[i] == 0)

{

XoaVT(a, n, i);

i--;

}

}

}

void ThemPT\_SauPTFirst (int \*a, int &n, int &x)

{

printf("\nNhap PT x: "); scanf\_s("%d", &x);

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

{

a[i + 1] = a[i];

}

a[1] = x;

n++;

}

int kiemtrascp (int n)

{

if (sqrt(float(n)) == (int)sqrt(float(n)))

return 1;

else

return 0;

}

int TongSCP (int \*a, int n)

{

int s = 0;

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

{

if (kiemtrascp(a[i]) == 1)

{

s += a[i];

}

}

return s;

}

void XuatPT\_CucDai (int \*a, int n)

{

int dem = 0;

printf("Xuat cac PT cuc dai trong mang:\n");

if (a[0] > a[1])

{

dem++;

printf("%5d", a[0]);

}

if (a[n - 1] > a[n - 2])

{

dem++;

printf("%5d", a[n - 1]);

}

for (int i = 1; i < n - 1; i++)

{

if (a[i] > a[i - 1] && a[i] > a[i + 1])

{

dem++;

printf("%5d", a[i]);

}

}

printf("\n");

if (dem == 0)

{

printf("\t- Khong co PT nao trong mang la so cuc dai\n");

}

}

void XuatPT\_CucTieu (int \*a, int n)

{

int dem = 0;

printf("Xuat cac PT cuc tieu trong mang:\n");

if (a[0] < a[1])

{

dem++;

printf("%5d", a[0]);

}

if (a[n - 1] < a[n - 2])

{

dem++;

printf("%5d", a[n - 1]);

}

for (int i = 1; i < n - 1; i++)

{

if (a[i] < a[i - 1] && a[i] < a[i + 1])

{

dem++;

printf("%5d", a[i]);

}

}

printf("\n");

if (dem == 0)

{

printf("\t- Khong co PT nao trong mang la so cuc tieu\n");

}

}

void XoaPT (int \*a, int &n, int &x)

{

int dem = 0;

printf("\nNhap PT x: "); scanf\_s("%d", &x);

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

{

if (a[i] == x)

{

dem++;

XoaVT(a, n, i);

i--;

}

}

if (dem == 0)

{

printf("Khong co PT %d trong mang\n", x);

}

}

void themPT (int \*a, int &n, int &x, int &vitri)

{

printf("\nNhap PT x: "); scanf\_s("%d", &x);

printf("\nNhap vi tri k: "); scanf\_s("%d", &vitri);

if (vitri < 0)

{

for (int i = n - 1; i >= vitri; i--)

{

a[i + 1] = a[i];

}

a[0] = x;

n++;

}

else if (vitri > n)

{

a[n] = x;

n++;

}

else

{

for (int i = n - 1; i >= vitri; i--)

{

a[i + 1] = a[i];

}

a[vitri] = x;

n++;

}

}

int kiemtrachanle (int \*a, int n)

{

int dem = 0;

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

if ((a[i] % 2 == 0 && a[i + 1] % 2 != 0) || (a[i] % 2 != 0 && a[i + 1] % 2 == 0))

dem++;

return dem;

}

void swap (int &x, int &y)

{

int temp = x;

x = y;

y = temp;

}

void DuaChanLenDau\_LeXuongDuoi (int \*a, int n)

{

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

{

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

{

if (a[j] % 2 == 0)

{

swap(a[i], a[j]);

break;

}

}

}

}

Bài 8 và Bài 12:

#define \_CRT\_NONSTDC\_NO\_WARNINGS

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <conio.h>

#include <stdlib.h>

#include <math.h>

#define MAX 100

struct PS

{

int ts;

int ms;

};

void nhap1PS (PS \*ps);

void xuat1PS (PS \*ps);

void nhapN (int &n);

void nhapdsPS (PS \*dsPS, int n);

void xuatdsPS (PS \*dsPS, int n);

void PT\_MauLonHonTu (PS \*dsPS, int n);

int DemPS\_Chan (PS \*dsPS, int n);

int ucln (PS \*dsPS);

void RutgonPS (PS \*dsPS, int n);

void rutgon1PS (PS \*dsPS);

void tichPS (PS \*dsPS, int n);

void timPSmax\_min (PS \*dsPS, int n);

int Tim\_VT\_k (PS \*dsPS, int n, PS k);

void XoaPT\_TaiK (PS \*dsPS, int &n, PS &k);

void ThemPTx\_TaiK (PS \*dsPS, int &n, PS &x, int &k);

int main ()

{

int n, chon;

PS \*dsPS;

do

{

printf("1.Nhap/Xuat pt trong mang\n");

printf("2.Xuat cac phan so mau > tu\n");

printf("3.Dem cac phan so mau va tu chan\n");

printf("4.Rut gon phan so\n");

printf("5.Tinh tich cac PS\n");

printf("6.Tim phan tu lon nhat va nho nhat\n");

printf("7.Xoa pt tai vi tri k\n");

printf("8.Them PT x tai vi tri k\n");

printf("Chon chuc nang: "); scanf\_s("%d", &chon);

switch (chon)

{

case 1:

{

int choose;

do

{

printf("1.Nhap pt trong mang\n");

printf("2.Xuat pt trong mang\n");

printf("Chon chuc nang: "); scanf\_s("%d", &choose);

switch (choose)

{

case 1:

{

nhapN(n);

dsPS = (PS \*)malloc(n\*sizeof(PS));

nhapdsPS(dsPS, n);

break;

}

case 2:

{

xuatdsPS(dsPS, n);

break;

}

}

}

while (choose != 1 && choose != 2);

break;

}

case 2:

{

PT\_MauLonHonTu(dsPS, n);

break;

}

case 3:

{

printf("So luong phan so co mau va tu chan trong mang la: %d\n", DemPS\_Chan(dsPS, n));

}

case 4:

{

RutgonPS(dsPS, n);

break;

}

case 5:

{

tichPS(dsPS, n);

break;

}

case 6:

{

timPSmax\_min(dsPS, n);

break;

}

case 7:

{

PS k;

XoaPT\_TaiK(dsPS, n, k);

break;

}

case 8:

{

PS x;

int k;

ThemPTx\_TaiK(dsPS, n, x, k);

break;

}

}

}

while (chon != 0);

return 0;

getch ();

}

void nhap1PS (PS \*ps)

{

printf("\tNhap tu so: "); scanf\_s("%d", &ps->ts);

do

{

try

{

printf("\tNhap mau so: "); scanf\_s("%d", &ps->ms);

if (ps->ms == 0)

{

throw "\t- Vui long nhap mau khac 0";

}

}

catch (const char \*msg)

{

printf("%s\n", msg);

}

}

while (ps->ms == 0);

}

void xuat1PS (PS \*ps)

{

if (ps->ts < 0 && ps->ms < 0)

{

printf("\tPhan so: %d/%d\n", ps->ts \* -1, ps->ms \* -1);

}

else if (ps->ms < 0)

{

printf("\tPhan so: %d/%d\n", ps->ts \* -1, ps->ms \* -1);

}

else if (ps->ms == 1)

{

printf("\tPhan so: %d\n", ps->ts);

}

else

printf("\tPhan so: %d/%d\n", ps->ts, ps->ms);

}

void nhapN (int &n)

{

printf("\tNhap danh sach phan so\n");

printf("Nhap so luong phan so: "); scanf\_s("%d", &n);

}

void nhapdsPS (PS \*dsPS, int n)

{

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

{

printf("Nhap phan so thu %d\n", i + 1);

nhap1PS(dsPS + i);

}

}

void xuatdsPS (PS \*dsPS, int n)

{

printf("Danh sach phan so vua nhap la: \n");

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

{

printf("- Phan so thu %d:\n", i + 1);

xuat1PS(dsPS + i);

printf("\t----------------------\n");

}

}

void PT\_MauLonHonTu (PS \*dsPS, int n)

{

printf("Cac phan so co mau > tu:\n");

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

{

if ((dsPS + i)->ms > (dsPS + i)->ts)

{

printf("\t%d/%d\n", (dsPS + i)->ts, (dsPS + i)->ms);

}

}

}

int DemPS\_Chan (PS \*dsPS, int n)

{

int dem = 0;

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

{

if ((dsPS + i)->ts % 2 == 0 && (dsPS + i)->ms % 2 == 0)

{

dem++;

}

}

return dem;

}

int ucln (PS \*dsPS)

{

int a = dsPS->ts;

int b = dsPS->ms;

a = abs(a);

b = abs(b);

int USCLN = 1;

if (a == 0 || b == 0)

{

USCLN = (a + b);

}

else

{

while (a != b)

{

if (a > b)

{

a = a - b;

}

else

{

b = b - a;

}

}

USCLN = a;

}

return USCLN;

}

void RutgonPS(PS \*dsPS, int n)

{

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

{

if((dsPS + i)->ts != 0)

{

int u = ucln((dsPS + i));

(dsPS + i)->ts = (dsPS + i)->ts / u;

(dsPS + i)->ms = (dsPS + i)->ms / u;

}

}

}

void rutgon1PS (PS \*dsPS)

{

int u = ucln(dsPS);

dsPS->ts = dsPS->ts / u;

dsPS->ms = dsPS->ms / u;

printf("%d/%d\n", dsPS->ts, dsPS->ms);

}

void tichPS (PS \*dsPS, int n)

{

PS tich = dsPS[0];

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

{

tich.ts = tich.ts \* (dsPS + i)->ts;

tich.ms = tich.ms \* (dsPS + i)->ms;

}

printf("Tich cac phan so la:\n");

xuat1PS(&tich);

printf("Phan so %d/%d sau khi rut gon la: ", tich.ts, tich.ms);

if (tich.ts == 0)

{

printf("0\n");

}

else

{

rutgon1PS(&tich);

}

}

void timPSmax\_min (PS \*dsPS, int n)

{

PS max = dsPS[0];

PS min = dsPS[0];

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

{

if (((float)(dsPS + i)->ts / (dsPS + i)->ms) > (float)max.ts / max.ms)

{

max = dsPS[i];

}

}

if (max.ts < 0 && max.ms < 0)

{

printf("Phan so lon nhat trong danh sach la: %d/%d\n", max.ts \* -1, max.ms \* -1);

}

else if (max.ms < 0)

{

printf("Phan so lon nhat trong danh sach la: %d/%d\n", max.ts \* -1, max.ms \* -1);

}

else if (max.ms == 1)

{

printf("Phan so lon nhat trong danh sach la: %d\n", max.ts);

}

else

{

printf("Phan so lon nhat trong danh sach la: %d/%d\n", max.ts, max.ms);

}

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

{

if (((float)(dsPS + i)->ts) / (dsPS + i)->ms < (float)min.ts / min.ms)

{

min = dsPS[i];

}

}

if (min.ts < 0 && min.ms < 0)

{

printf("Phan so nho nhat trong danh sach la: %d/%d\n", min.ts \* -1, min.ms \* -1);

}

else if (min.ms < 0)

{

printf("Phan so nho nhat trong danh sach la: %d/%d\n", min.ts \* -1, min.ms \* -1);

}

else if (min.ms == 1)

{

printf("Phan so nho nhat trong danh sach la: %d\n", min.ts);

}

else

{

printf("Phan so nho nhat trong danh sach la: %d/%d\n", min.ts, min.ms);

}

}

int Tim\_VT\_k (PS \*dsPS, int n, PS k)

{

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

{

if ((dsPS + i)->ts == k.ts && (dsPS + i)->ms == k.ms)

{

return i;

break;

}

}

return -1;

}

void XoaPT\_TaiK (PS \*dsPS, int &n, PS &k)

{

printf("Nhap phan so muon xoa: \n");

printf("\tNhap tu so: "); scanf\_s("%d", &k.ts);

printf("\tNhap mau so: "); scanf\_s("%d", &k.ms);

int vitri = Tim\_VT\_k(dsPS, n, k);

if (vitri == -1)

{

printf("Khong tim thay phan so %d/%d muon xoa\n", k.ts, k.ms);

}

else

{

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

{

dsPS[i].ts = dsPS[i + 1].ts;

dsPS[i].ms = dsPS[i + 1].ms;

}

n--;

}

}

void ThemPTx\_TaiK (PS \*dsPS, int &n, PS &x, int &k)

{

printf("Nhap phan so muon them: \n");

printf("\tNhap tu so: "); scanf\_s("%d", &x.ts);

printf("\tNhap mau so: "); scanf\_s("%d", &x.ms);

printf("Vi tri can them: "); scanf\_s("%d", &k);

if (k - 1 < 0)

{

for (int i = n - 1; i >= k; i--)

{

dsPS[i + 1] = dsPS[i];

}

dsPS[0] = x;

n++;

}

else if (k - 1 > n)

{

dsPS[n] = x;

n++;

}

else

{

for (int i = n - 1; i >= k - 1; i--)

{

dsPS[i + 1] = dsPS[i];

}

dsPS[k - 1] = x;

n++;

}

}

Bài 10:

#define \_CRT\_NONSTDC\_NO\_WARNINGS

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <conio.h>

#include <string.h>

#include <stdlib.h>

void NhapChuoi (char \*s);

void XuatChuoi (char \*s);

void XuatTungKyTu (char \*s);

void ChuyenKyTuHoa (char \*s);

void VietHoaChu\_SauDauCach (char \*s);

int main ()

{

char \*s;

s = (char \*)malloc(100\*sizeof(char));

int chon;

do

{

printf("1.Nhap chuoi\n");

printf("2.Xuat chuoi\n");

printf("3.Xuat gia tri tung ky tu\n");

printf("4.Chuyen cac ky tu ve chu hoa\n");

printf("5.Chuyen cac ky tu dau moi tu (dung sau dau cach) ve chu hoa\n");

printf("Chon chuc nang: "); scanf\_s("%d", &chon);

switch (chon)

{

case 1:

{

NhapChuoi(s);

break;

}

case 2:

{

XuatChuoi(s);

break;

}

case 3:

{

XuatTungKyTu(s);

break;

}

case 4:

{

ChuyenKyTuHoa(s);

break;

}

case 5:

{

VietHoaChu\_SauDauCach(s);

break;

}

}

}

while (chon != 0);

return 0;

getch ();

free(s);

}

void NhapChuoi (char \*s)

{

printf("Nhap chuoi s: ");

fflush(stdin);

gets(s);

}

void XuatChuoi (char \*s)

{

printf("Xuat chuoi s: ");

puts(s);

}

void XuatTungKyTu (char \*s)

{

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

{

printf("Ky tu s[%d]: %c\n", i, \*(s + i));

}

}

void ChuyenKyTuHoa (char \*s)

{

printf("Chuyen sang chu hoa: %s\n", strupr(s));

}

void VietHoaChu\_SauDauCach (char \*s)

{

if (\*s >= 'a' && \*s <= 'z')

{

\*s -= 32;

}

for (int i = 1; i < strlen(s); i++)

{

if (\*(s + i - 1) == ' ' && \*(s + i) >= 'a' && \*(s + i) <= 'z')

{

\*(s + i) -= 32;

}

}

}

Bài 13:

#define \_CRT\_NONSTDC\_NO\_WARNINGS

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <conio.h>

#include <stdlib.h>

#include <math.h>

#include <time.h>

#include <string.h>

void nhapM1CA\_Random (int \*&a, int &na);

void nhapM1CB\_Random (int \*&b, int &nb);

void xuatM1C(int \*a, int n);

void Chia\_PTMangA\_Cho\_PTMangB (int \*a, int na, int \*b, int nb);

int main ()

{

int \*a, \*b;

int na, nb;

nhapM1CA\_Random(a, na);

xuatM1C(a, na);

nhapM1CB\_Random(b, nb);

xuatM1C(b, nb);

Chia\_PTMangA\_Cho\_PTMangB(a, na, b, nb);

return 0;

getch ();

}

void nhapM1CA\_Random (int \*&a, int &na)

{

printf("Nhap so luong PT mang A: "); scanf("%d", &na);

a = (int \*)malloc(na\*sizeof(int));

srand((unsigned)time(NULL));

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

{

\*(a + i) = rand() % 100;

}

}

void nhapM1CB\_Random (int \*&b, int &nb)

{

printf("Nhap so luong PT mang B: "); scanf("%d", &nb);

b = (int \*)malloc(nb\*sizeof(int));

srand((unsigned)time(NULL));

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

{

\*(b + i) = rand() % 100;

}

}

void xuatM1C(int \*a, int n)

{

printf("Xuat mang:\n");

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

{

printf("%5d", a[i]);

}

printf("\n");

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

{

printf("\nPhan tu thu %d co gia tri %d va dia chi o nho la %x\n", i, \*(a + i), a + i);

}

}

void Chia\_PTMangA\_Cho\_PTMangB (int \*a, int na, int \*b, int nb)

{

float \*ab;

ab = (float \*)malloc(nb\*sizeof(float));

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

{

printf("PT %d cua mang A chia cho PT cua mang B:\n", \*(a + i));

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

{

if (b[j] == 0)

{

printf("\t%d : %d = %s\n", \*(a + i), \*(b + j), "Loi chia cho 0");

}

else

{

\*(ab + j) = (float)\*(a + i) / \*(b + j);

printf("\t%d : %d = %.3f\n", \*(a + i), \*(b + j), \*(ab + j));

}

}

}

}