#include <iostream>

#include <clocale>

using namespace std;

int\* Perechislenie(int mM);

int\* Talk\_A(int mM);

int\* Talk\_B(int mM);

void neto();

void mn(int \*A, int mA, int \*B, int mB);

int\* edin(int \*A, int \*B, int mA, int mB);

int\* peresechenie(int \*A, int \*B, int mA, int mB);

int\* Pa3HocTb(int \*U, int \*W, int mU, int mW);

int\* Add(int \*M, int mM);

int\* SimmRaz(int \*A, int \*B, int mA, int mB);

int\* Dekart(int \*A, int \*B, int mA, int mB);

int main()

{

setlocale(LC\_ALL, "rus");

int \*A, \*B;

int mA, mB, mC, mD, mR, mY, mE, mP, mS;

int i, Operaziya = 0, ZMA=0, ZMB=0;

cout << "Введите мощность множества А (от 0 до 100): ";

cin >> mA;

A = new int[mA];

cout << endl;

cout << "Введите мощность множества B (от 0 до 100): ";

cin >> mB;

B = new int[mB];

cout << endl << "--------------------------------------------------------------------------------------" << endl << endl;

cout << "Выберите способ задания множества А: " << endl;

cout << " 1. Перечисление.\n 2. Высказывание.\n";

cin >> ZMA;

switch (ZMA)

{

case 1:

A = Perechislenie(mA);

break;

case 2:

A = Talk\_A(mA);

break;

default:

neto();

return 0;

}

cout << endl << endl;

cout << "Выберите способ задания множества B: " << endl;

cout << " 1. Перечисление.\n 2. Высказывание.\n";

cin >> ZMB;

switch (ZMB)

{

case 1:

B = Perechislenie(mB);

break;

case 2:

B = Talk\_B(mB);

break;

default:

neto();

return 0;

}

cout << endl << endl << "Множества А и В заполнены!" << endl << endl;

while (true)

{

system("pause");

system("cls");

mn(A, mA, B, mB);

cout << "Выберите операцию:" << endl;

cout << " 1. Объединение множеств\n 2. Пересечение множеств\n 3. Разность A и В\n 4. Разность B и A\n 5. Дополнение А до U\n 6. Дополнение B до U\n 7. Симметричная разность A u B\n 8. Декартово произведение A на B\n 9. Декартово произведение B на A\n 0. Выход\n";

cin >> Operaziya;

system("cls");

switch (Operaziya)

{

case 1:

{

cout << "Объединение множеств" << endl;

int \*C = new int[mA + mB];

C = edin(A, B, mA, mB);

mC = C[0];

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

cout << C[i] << " ";

cout << endl;

}

break;

case 2:

{

cout << "Пересечение множеств" << endl;

int \*D = new int[mA + mB];

D = peresechenie(A, B, mA, mB);

mD = D[0];

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

cout << D[i] << " ";

cout << endl;

}

break;

case 3:

{

cout << "Разность A и В" << endl;

int \*R = new int[mA];

R = Pa3HocTb(A, B, mA, mB);

mR = R[0];

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

cout << R[i] << " ";

cout << endl;

}

break;

case 4:

{

cout << "Разность B и A" << endl;

int \*Y = new int[mB];

Y = Pa3HocTb(B, A, mB, mA);

mY = Y[0];

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

cout << Y[i] << " ";

cout << endl;

}

break;

case 5:

{

cout << "Дополнение А до U" << endl;

int \*E = new int[1001];

E = Add(A, mA);

mE = E[0];

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

cout << E[i] << " ";

cout << endl;

}

break;

case 6:

{

cout << "Дополнение B до U" << endl;

int \*E = new int[1001];

E = Add(B, mB);

mE = E[0];

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

cout << E[i] << " ";

cout << endl;

}

break;

case 7:

{

cout << "Симметричная разность А и В" << endl;

int \*S = new int[mA + mB + 2];

S = SimmRaz(A, B, mA, mB);

mS = S[0];

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

cout << S[i] << " ";

cout << endl;

}

break;

case 8:

{

cout << "Декартово произведение A на B" << endl;

int \*P = new int[2 \* (mA \* mB) + 2];

P = Dekart(A, B, mA, mB);

mP = P[0];

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

{

cout << "<" << P[i] << ", " << P[i + 1] << ">";

if (i < mP - 1)

cout << ", ";

i++;

}

cout << endl;

}

break;

case 9:

{

cout << "Декартово произведение B на A" << endl;

int \*P = new int[2 \* (mA \* mB) + 2];

P = Dekart(B, A, mB, mA);

mP = P[0];

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

{

cout << "<" << P[i] << ", " << P[i + 1] << ">";

if (i < mP - 1)

cout << ", ";

i++;

}

cout << endl;

}

break;

case 0:

return 0;

}

}

}

int\* Perechislenie(int mM)

{

int \*M = new int[mM];

if (mM == 0)

{

cout << endl << "Пустое множество!" << endl;

return M;

}

cout << endl << "Введте элементы множества: " << endl;

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

cin >> M[i];

return M;

}

int\* Talk\_A (int n)

{

int \*A = new int[n];

int a=0;

cout << endl << "Элементы множества А: ";

if (n == 0)

{

cout << endl << endl << "Пустое множество!";

return A;

}

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

{

a = 4 \* (x + 1) + 2 \* (n - x);

A[x] = a;

cout << endl << a << " ";

}

return A;

}

int\* Talk\_B(int m)

{

int \*B = new int[m];

int b = 0;

cout << endl << "Элементы множества B: ";

if (m == 0)

{

cout << endl << endl << "Пустое множество!";

return B;

}

for (int y = 0; y < m; y++)

{

b = 2 \* (m - y) + 3 \* y;

B[y] = b;

cout << endl << b << " ";

}

return B;

}

void neto()

{

cout << endl << endl << endl << " Sirius li? Там нет такой цифры. Код привередливый, код сломан!!!\n ПЕРЕзапускай ПроГРаМММУУУУУ.....\n EROoorr........Ereoreorr............" << endl << endl << endl;

system("pause");

}

void mn(int \*A, int mA, int \*B, int mB)

{

cout << "A = ";

cout << "{";

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

{

cout << A[i];

if (i < (mA - 1))

cout << ", ";

}

cout << "}" << endl;

cout << "B = ";

cout << "{";

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

{

cout << B[i];

if (i < (mB - 1))

cout << ", ";

}

cout << "}" << endl;

}

int\* edin(int \*A, int \*B, int mA, int mB)

{

int mO = mA + mB + 1, i = 0;

int \*C = new int[mO];

if (mB == 0 && mA == 0)

{

cout << "Пустое множество!";

return C;

}

if (mA == 0)

{

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

{

C[i + 1] = B[i];

}

C[0] = i;

return C;

}

if (mB == 0)

{

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

{

C[i + 1] = A[i];

}

C[0] = i;

return C;

}

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

{

C[i + 1] = A[i];

}

for (int b = 0; b < mB; b++)

{

for (int a = 0; a < mA; a++)

{

if (B[b] == A[a])

break;

if (B[b] != A[a])

{

if (a == (mA - 1))

{

i++;

C[i] = B[b];

}

else continue;

}

}

}

C[0] = i;

return C;

}

int\* peresechenie(int \*A, int \*B, int mA, int mB)

{

int \*D = new int[mB + mA];

int i = 0;

if (mA == 0 || mB == 0)

{

cout << "Пустое множество!";

return D;

}

for (int b = 0; b < mB; b++)

{

for (int a = 0; a < mA; a++)

{

if (B[b] == A[a])

{

i++;

D[i] = B[b];

}

}

}

if (i==0)

cout << endl << "Пустое множество!" << endl;

D[0] = i;

return D;

}

int\* Pa3HocTb(int \*U, int \*W, int mU, int mW)

{

int \*R = new int[mU+1];

int r = 0;

if (mU == 0)

{

cout << "Пустое множество!";

return R;

}

if (mW == 0)

{

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

{

R[i+1] = U[i];

}

R[0] = mU;

return R;

}

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

{

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

{

if (U[i] == W[j])

break;

if (j == (mW - 1))

{

r++;

R[r] = U[i];

}

}

}

if (r == 0)

cout << "Пустое множество!";

R[0] = r;

return R;

}

int\* Add(int \*M, int mM)

{

int \*U = new int[1000];

for (int u = 1; u <= 1000; u++)

U[u] = u;

int \*E = new int[1001];

int e = 0;

if (mM == 0)

{

U[0] = 1000;

return U;

}

for (int i = 1; i < 1001; i++)

{

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

{

if (U[i] == M[j])

break;

if (U[i] != M[j])

{

if (j == (mM - 1))

{

e++;

E[e] = U[i];

}

else continue;

}

}

}

E[0] = e;

return E;

}

int\* SimmRaz(int \*A, int \*B, int mA, int mB)

{

int \*S = new int[mA+mB+2];

int s = 0, o=0;

if (mA == 0 && mB == 0)

{

cout << "Пустое множество!";

return S;

}

if (mA == 0)

{

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

{

S[i + 1] = B[i];

}

S[0] = mB;

return S;

}

if (mB == 0)

{

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

{

S[i + 1] = A[i];

}

S[0] = mA;

return S;

}

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

{

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

{

o = 0;

if (A[i] == B[j])

continue;

for (int l = 1; l <= s; l++)

{

if (S[l] == A[i])

{

o = 1;

break;

}

}

for (int v = 0; v < mB; v++)

{

if (A[i] == B[v])

{

o = 1;

break;

}

}

if (o == 1)

continue;

s++;

S[s] = A[i];

}

}

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

{

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

{

o = 0;

for (int l = 1; l <= s; l++)

{

if (S[l] == B[j])

{

o = 1;

break;

}

}

for (int v = 0; v < mA; v++)

{

if (A[v] == B[j])

{

o = 1;

break;

}

}

if (o == 1)

continue;

s++;

S[s] = B[j];

}

}

if (s == 0)

cout << "Пустое множество!";

S[0] = s;

return S;

}

int\* Dekart(int \*A, int \*B, int mA, int mB)

{

int \*P = new int[2 \* (mA\*mB) + 2];

int p = 0, o=0;

if (mA < 1||mB<1)

{

cout << "Пустое множество!";

return P;

}

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

{

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

{

o = 0;

for (int k = 1; k < p; k+=2)

{

if (A[i] == P[k] && B[j] == P[k + 1])

{

o = 1;

break;

}

}

if (o == 1)

continue;

p++;

P[p] = A[i];

p++;

P[p] = B[j];

}

}

P[0] = p;

return P;

}