#include <iostream>

using namespace std;

class vecTor

{

public:

vecTor() {}

int\* push\_back(int number)

{

if (size == 0)

{

mass = new int[size + 1];

}

else

{

int\* temp = new int[size + 1];

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

{

temp[i] = mass[i];

}

delete[] mass;

mass = temp;

}

mass[size] = number;

size++;

return mass;

}

int sizeOfMass()

{

return size;

}

void append(vecTor& other)

{

int newSize = this->size + other.size;

int\* temp = new int[newSize];

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

{

if (i<this->size)

{

temp[i] = this->mass[i];

}

else

{

temp[i] = other.mass[i - this->size];

}

}

delete[] this->mass;

this->size = newSize;

this->mass = temp;

}

int\* clear()

{

delete[] mass;

size = 0;

return mass;

}

int\* pop\_back()

{

int\* temp = new int[size - 1];

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

{

temp[i] = mass[i];

}

delete[] mass;

mass = temp;

size--;

return mass;

}

bool empty()

{

if (size == 0)

return true;

else

return false;

}

void Show()

{

cout << "\n";

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

{

cout << mass[i] << " ";

}

}

int get\_at(int a)

{

return mass[a];

}

private:

int size = 0;

int\* mass = nullptr;

};

void ShowObj(vecTor& obj)

{

cout << "by func\n";

obj.Show();

}

int main()

{

vecTor mass;

mass.push\_back(6);

mass.push\_back(4);

mass.push\_back(5);

mass.push\_back(7);

mass.push\_back(47);

cout << "mass: ";

mass.Show();

vecTor mass2;

mass2.push\_back(5);

mass2.push\_back(54);

mass2.push\_back(5);

mass2.push\_back(4);

mass2.push\_back(6);

cout << "\nmass2: ";

mass2.Show();

cout << "\n\nafter apppend";

mass.append(mass2);

mass.Show();

}