#include<iostream>

#include<string>

#include<cmath>

using namespace std;

class num {

private:

int\* arr;

int size = 0;

int a;

public:

num(int n) {

a = n;

}

num() {

a = 0;

}

num(const num& other) {

this->arr = new int[other.size];

this->size = other.size;

for (int i = 0; i < other.size; i++) {

this->arr[i] = other.arr[i];

}

}

~num() {

delete arr;

}

void setA(double value) {

a = value;

}

double getA() {

return a;

}

num& operator = (const num& other) {

this->size = other.size;

if (this->arr != NULL) {

delete[] this->arr;

}

this->arr = new int[other.size];

for (int i = 0; i < other.size; i++) {

this->arr[i] = other.arr[i];

}

return \*this;

}

num& operator = (const num& other) {

this->a = other.a;

this->size = other.size;

if (this->arr != nullptr) {

delete[] this->arr;

}

this->arr = new int[other.size];

for (int i = 0; i < other.size; i++) {

this->arr[i] = other.arr[i];

}

return \*this;

}

num operator + (const num & other) {

num temp;

temp.a = this->a + other.a;

return temp;

}

num operator - (const num& other) {

num temp;

temp.a = this->a - other.a;

return temp;

}

num operator \* (const num& other) {

num temp;

temp.a = this->a \* other.a;

return temp;

}

num operator / (const num& other) {

num temp;

temp.a = this->a / other.a;

return temp;

}

//////////////

void toBinary() {

int n = a;

do

{

push(arr, size, (n % 2));

n = n / 2;

} while (n > 0);

}

void toBinary(int n) {

do

{

push(arr, size, (n % 2));

n = n / 2;

} while (n > 0);

}

int toDec() {

int b = 0;

int s = 0;

for (int i = size-1; i >= 0; i--) {

if (arr[i] != 0) {

b += pow(2, s);

}

s++;

}

return b;

}

void push(int\*& arr, int& n, int elem) {

int\* newArr = new int[n + 1];

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

newArr[i] = arr[i];

}

newArr[n] = elem;

n++;

delete[] arr;

arr = newArr;

}

void showContent() {

cout << "binary: ";

for (int i = size-1; i>-1; i--) {

cout << arr[i];

}

cout << endl << "decade: " << a << endl;

}

};

void main() {

num a(5);

num b(4);

num c(6);

num arr[3];

arr[0] = a;

arr[1] = b;

arr[2] = c;

int temp, max = a.getA(), min = a.getA();

int n = 3;

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

temp += arr[i].getA();

if (arr[i].getA() < min)

min = arr[i].getA();

if (arr[i].getA() > max)

max = arr[i].getA();

arr[i].setA(arr[i].getA() \* 3);

}

cout << max - min << endl;

cout << temp / n << endl;

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

{

arr[i].showContent();

}

}