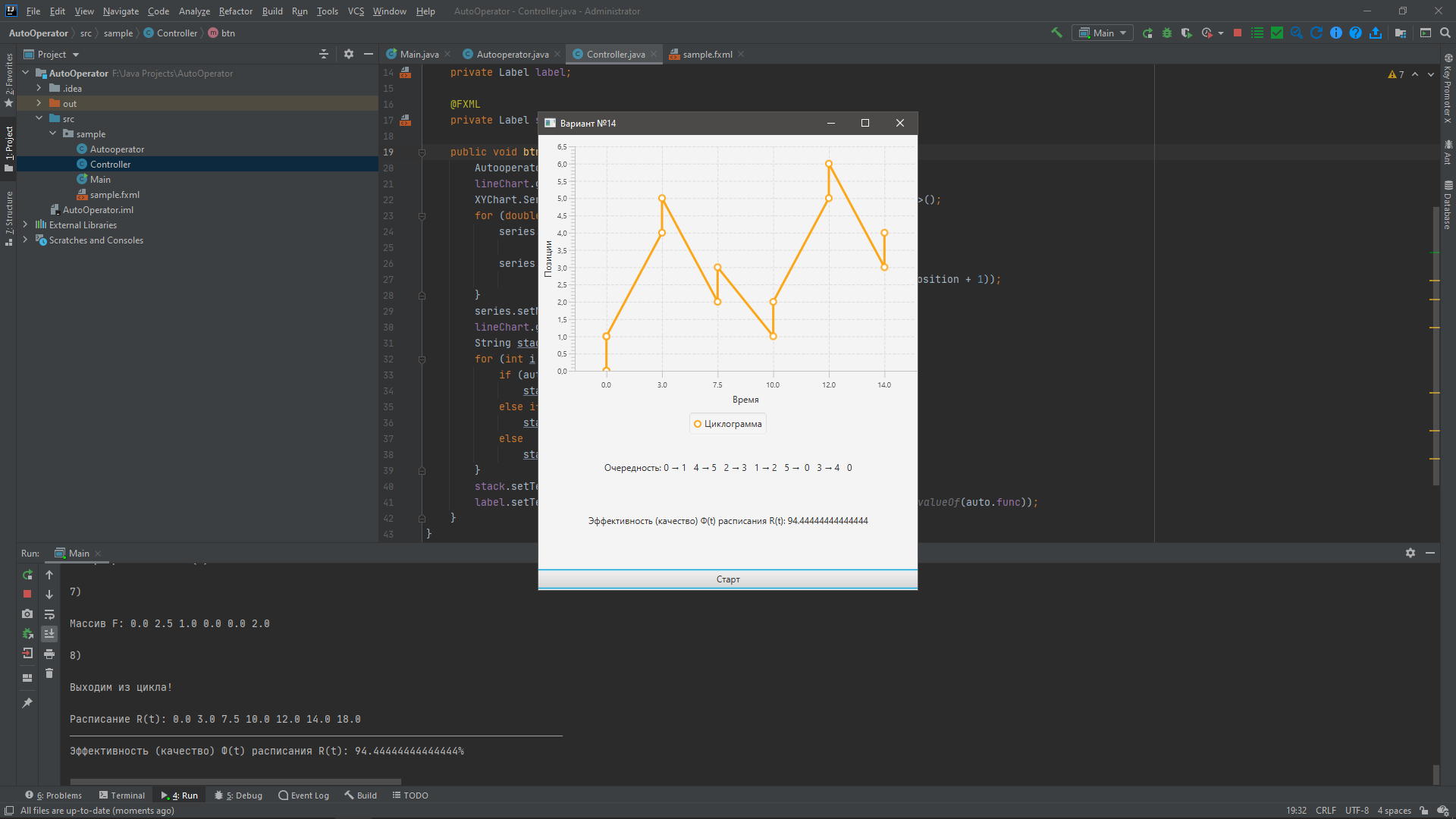
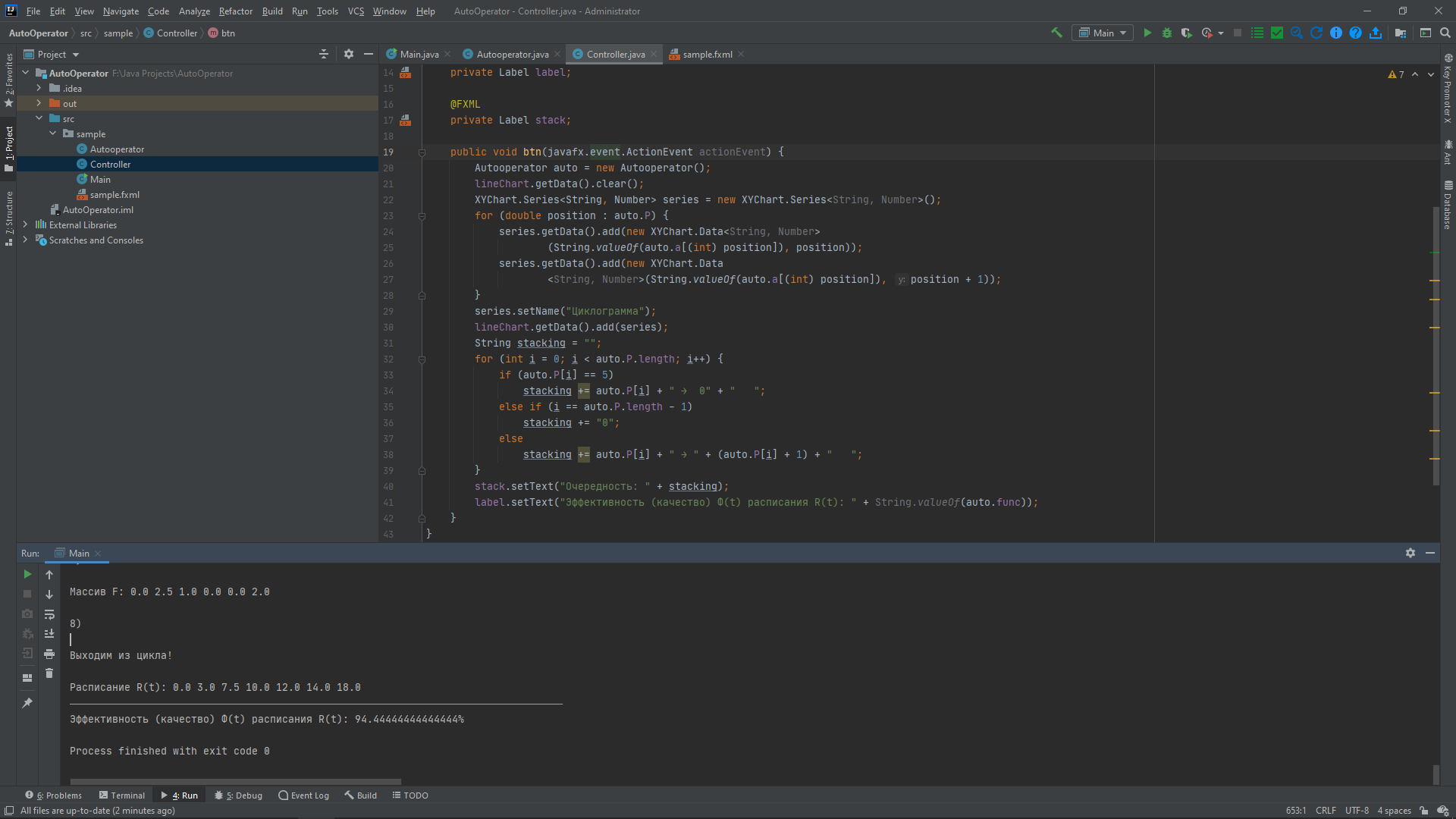
**ПРОТОТИП 0**

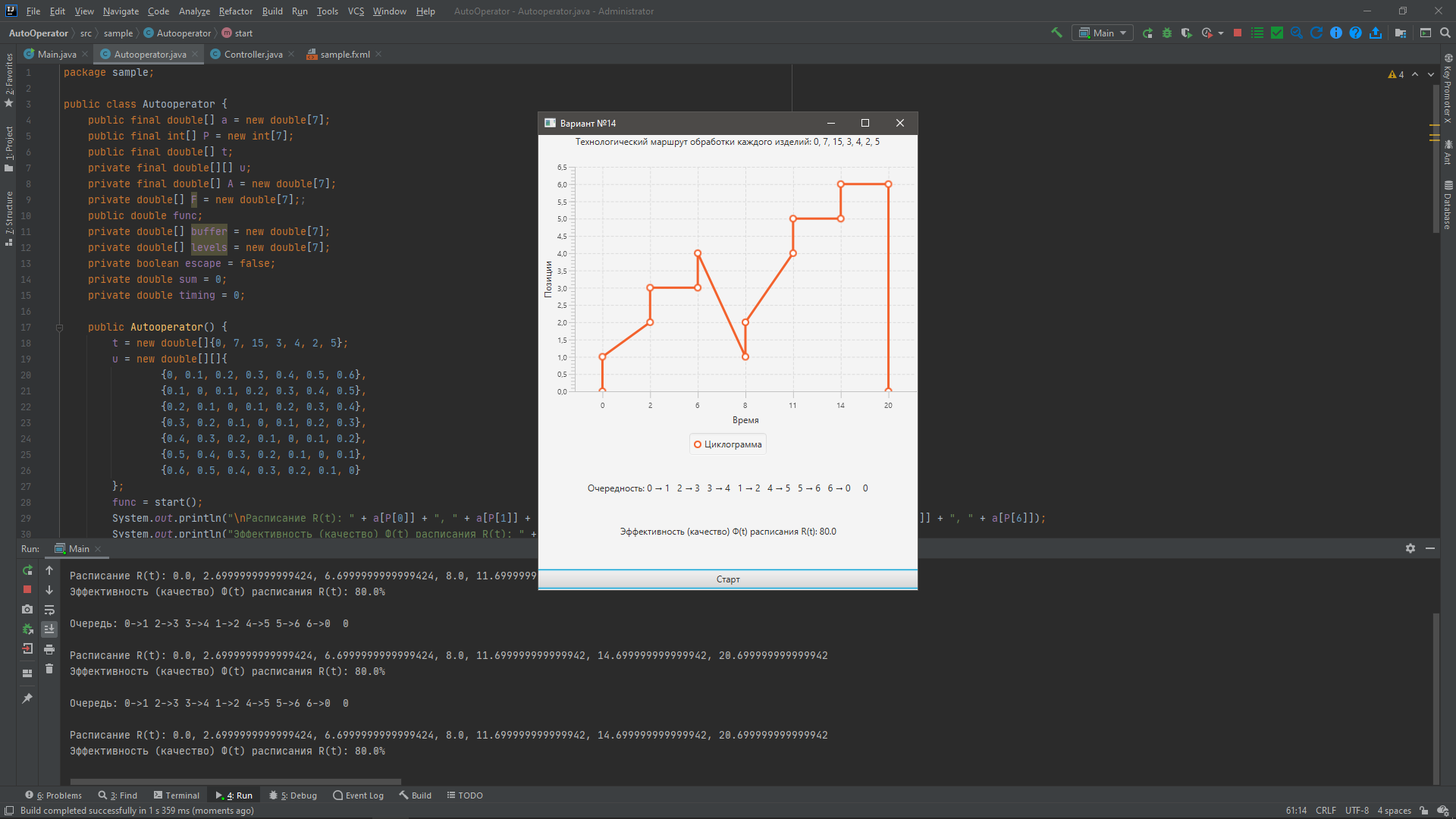
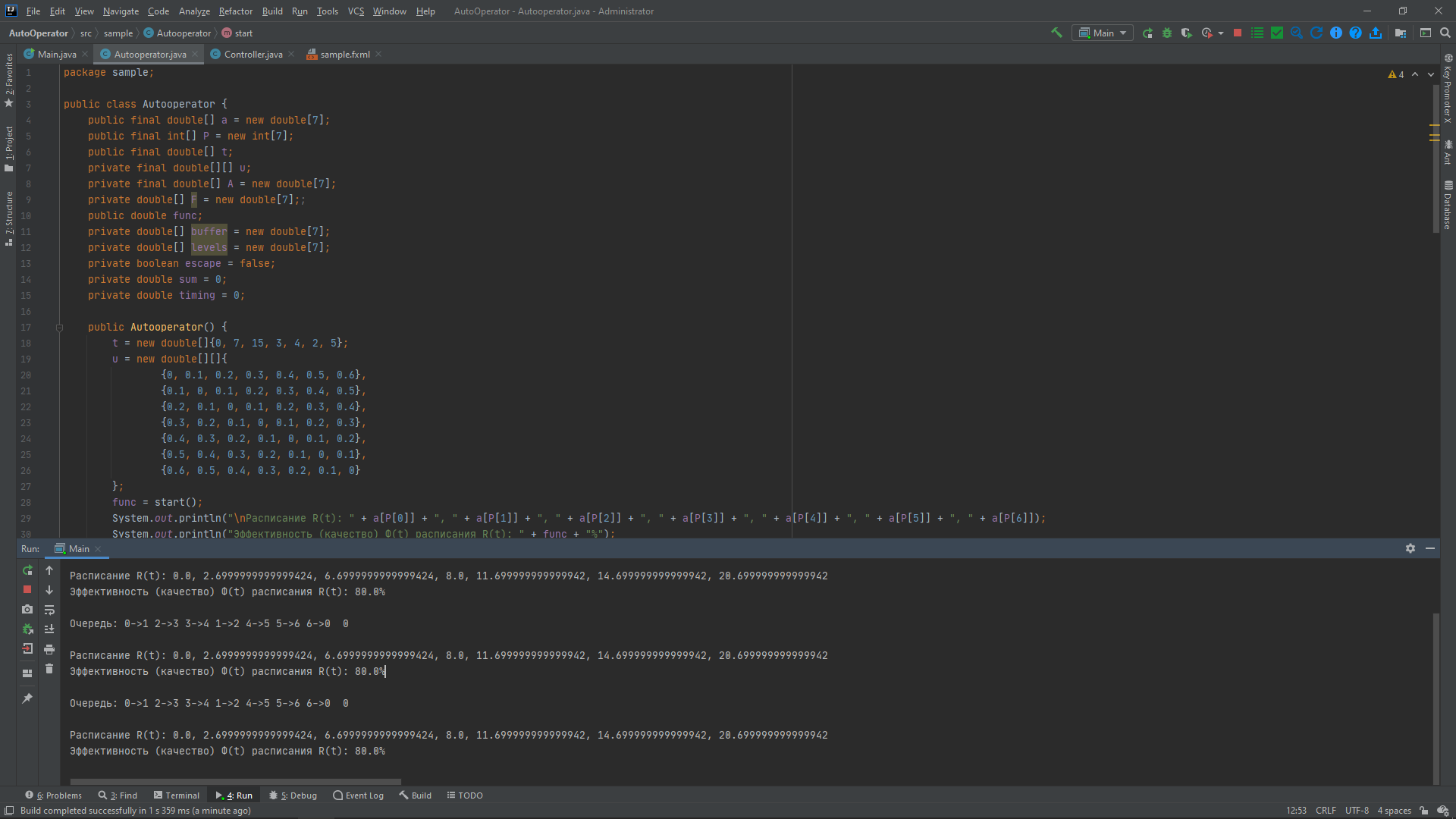




**КОД**

package sample;  
  
import java.util.Arrays;  
  
public class Autooperator {  
 private final double[] t;  
 private final int[] x;  
 private final double[][] u;  
 private final double[] A = new double[6];  
 public final double[] a = new double[6];  
 private final int[] S = new int[6];  
 public final int[] P = new int[7];  
 private final double[] F = new double[6];  
 public double[] R;  
 public double func;  
  
 public Autooperator() {  
 t = new double[] {0, 9, 14.5, 5.5, 6, 8};  
 x = new int[] {1, 1, 1, 1, 1, 1};  
 u = new double[][] {  
 {0, 0.3, 0.7, 1, 1, 2},  
 {0.3, 0, 0.3, 0.5, 2, 1.5},  
 {0.7, 0.3, 0, 0.5, 1, 1},  
 {1, 0.5, 0.5, 0, 0.3, 0.5},  
 {1, 2, 1, 0.3, 0, 0.5},  
 {2, 1.5, 1, 0.5, 0.5, 0},  
 };  
 func = start();  
 System.*out*.println("\nЭффективность (качество) Ф(t) расписания R(t): " + func + "%");  
 }  
  
 private double start() {  
 for(int i = 0; i < 50; i++) System.*out*.print("―");  
 // 1  
 System.*out*.println("\n1)\n");  
 double tee;  
 double max = 0, sum = 0;  
 int buffer = 0;  
 for(int i = 0; i < 6; i++) {  
 if (t[i] > max) {  
 max = t[i];  
 buffer = i;  
 }  
 }  
 max = t[buffer] + x[buffer] + x[buffer - 1] + u[buffer - 1][buffer + 1];  
 System.*out*.println("max: " + max);  
 for(int i = 0; i < 6; i++) {  
 sum += x[i];  
 }  
 System.*out*.println("sum: " + sum);  
 tee = Math.*max*(max, sum);  
 double first\_tee = tee;  
 System.*out*.println("tee: " + tee);  
  
 // 2  
 System.*out*.println("\n2)\n");  
 double bigTee = 0;  
 for(int i = 0; i < 6; i++) {  
 bigTee += x[i] + t[i];  
 }  
 System.*out*.println("bigTee: " + bigTee);  
  
 // 3  
 System.*out*.println("\n3)\n");  
 double z = bigTee / tee + 1;  
 System.*out*.println("z: " + (int) z);  
  
 // 4  
 System.*out*.println("\n4)\n");  
 A[0] = 0;  
 System.*out*.println("A[" + 0 + "]: " + A[0]);  
 for(int i = 1; i < 6; i++) {  
 A[i] = A[i - 1] + x[i - 1] + t[i];  
 System.*out*.println("A[" + i + "]: " + A[i]);  
 }  
  
  
 // 5  
 System.*out*.println("\n5)\n");  
 for(int i = 0; i < 6; i++) {  
 if (A[i] < tee) {  
 a[i] = A[i];  
 S[i] = 0;  
 }  
 else if (tee < A[i] && A[i] < 2 \* tee) {  
 a[i] = A[i] - tee;  
 S[i] = 1;  
 }  
 else if (2 \* tee < A[i] && A[i] < 3 \* tee) {  
 a[i] = A[i] - 2 \* tee;  
 S[i] = 2;  
 }  
 System.*out*.println("a[" + i + "]: " + a[i]);  
 }  
  
 // 6  
 System.*out*.println("\n6)\n");  
 System.*out*.print("Исходный массив (a): ");  
 for(double element : a)  
 System.*out*.print(element + " ");  
 double[] buff = new double[6];  
 System.*arraycopy*(a, 0, buff, 0, 6);  
 Arrays.*sort*(buff);  
 for(int i = 0; i < 6; i++) {  
 for(int j = 0; j < 6; j++) {  
 if (buff[i] == a[j]) {  
 P[i] = j;  
 }  
 }  
 }  
 P[6] = 0;  
 System.*out*.print("\nОтсортировал массив (buff): ");  
 for(double element : buff)  
 System.*out*.print(element + " ");  
 System.*out*.print("\nОтсортировал массив (P): ");  
 for(double element : P)  
 System.*out*.print(element + " ");  
  
 // 7  
 System.*out*.println("\n\n7)\n");  
 for(int i = 0; i < 6; i++) {  
 if (i < 4) F[i] = a[P[i + 1]] - a[P[i]] - x[P[i]] - u[P[i + 1]][P[i] + 1];  
 else if (i == 4) F[i] = a[P[i + 1]] - a[P[i]] - x[P[i]] - u[P[i + 1]][0];  
 else if (i == 5) F[i] = tee - a[P[i]] - x[P[i]] - u[P[i + 1]][P[i] + 1];  
 }  
 System.*out*.print("Массив F: ");  
 for(double element : F)  
 System.*out*.print(element + " ");  
  
 // 8  
 System.*out*.println("\n\n8)\n");  
 double diffTee = 0;  
 boolean escape = true;  
 for(int i = 0; i < 6; i++) {  
 if (F[i] < 0 && S[i - 1] < S[i + 1]) {  
 System.*out*.println("F[" + i + "]: " + F[i]);  
 System.*out*.println("S[" + (i - 1) + "]: " + S[i - 1]);  
 System.*out*.println("S[" + (i + 1) + "]: " + S[i + 1]);  
 diffTee = Math.*abs*(F[i]) / (S[i + 1] - S[i - 1]);  
 // 9  
 System.*out*.println("\n9)\n");  
 tee = tee + diffTee;  
 System.*out*.println("diffTee: " + diffTee);  
 System.*out*.println("tee: " + tee);  
 escape = false;  
 }  
 }  
  
 int count = 0;  
  
 while (!escape) {  
 count++;  
 System.*out*.println("\nВходим в цикл №" + count);  
 // 5  
 System.*out*.println("\n5)\n");  
 for(int i = 0; i < 6; i++) {  
 if (A[i] < tee) {  
 a[i] = A[i];  
 S[i] = 0;  
 }  
 else if (tee < A[i] && A[i] < 2 \* tee) {  
 a[i] = A[i] - tee;  
 S[i] = 1;  
 }  
 else if (2 \* tee < A[i] && A[i] < 3 \* tee) {  
 a[i] = A[i] - 2 \* tee;  
 S[i] = 2;  
 }  
 System.*out*.println("a[" + i + "]: " + a[i]);  
 }  
  
 // 6  
 System.*out*.println("\n6)\n");  
 System.*out*.print("Исходный массив (a): ");  
 for(double element : a)  
 System.*out*.print(element + " ");  
 buff = new double[6];  
 System.*arraycopy*(a, 0, buff, 0, 6);  
 Arrays.*sort*(buff);  
 for(int i = 0; i < 6; i++) {  
 for(int j = 0; j < 6; j++) {  
 if (buff[i] == a[j]) {  
 P[i] = j;  
 }  
 }  
 }  
 P[6] = 0;  
 System.*out*.print("\nОтсортировал массив (buff): ");  
 for(double element : buff)  
 System.*out*.print(element + " ");  
 System.*out*.print("\nОтсортировал массив (P): ");  
 for(double element : P)  
 System.*out*.print(element + " ");  
  
 // 7  
 System.*out*.println("\n\n7)\n");  
 for(int i = 0; i < 6; i++) {  
 if (i < 4) F[i] = a[P[i + 1]] - a[P[i]] - x[P[i]] - u[P[i + 1]][P[i] + 1];  
 else if (i == 4) F[i] = a[P[i + 1]] - a[P[i]] - x[P[i]] - u[P[i + 1]][0];  
 else if (i == 5) F[i] = tee - a[P[i]] - x[P[i]] - u[P[i + 1]][P[i] + 1];  
 }  
 System.*out*.print("Массив F: ");  
 for(double element : F)  
 System.*out*.print(element + " ");  
  
 // 8  
 System.*out*.println("\n\n8)\n");  
 escape = true;  
 for(int i = 0; i < 6; i++) {  
 if (F[i] < 0 && S[i - 1] < S[i + 1]) {  
 System.*out*.println("F[" + i + "]: " + F[i]);  
 System.*out*.println("S[" + (i - 1) + "]: " + S[i - 1]);  
 System.*out*.println("S[" + (i + 1) + "]: " + S[i + 1]);  
 diffTee = Math.*abs*(F[i]) / (S[i + 1] - S[i - 1]);  
 // 9  
 System.*out*.println("\n9)\n");  
 tee = tee + diffTee;  
 System.*out*.println("diffTee: " + diffTee);  
 System.*out*.println("tee: " + tee);  
 escape = false;  
 }  
 }  
 }  
 System.*out*.println("Выходим из цикла!\n");  
 R = new double[P.length];  
 for(int i = 0; i < P.length - 1; i++) {  
 R[i] = a[P[i]];  
 }  
 R[R.length - 1] = tee;  
 System.*out*.print("Расписание R(t): ");  
 for(double element : R)  
 System.*out*.print(element + " ");  
 System.*out*.println();  
 for(int i = 0; i < 50; i++) System.*out*.print("―");  
 return (first\_tee / tee) \* 100;  
 }  
}

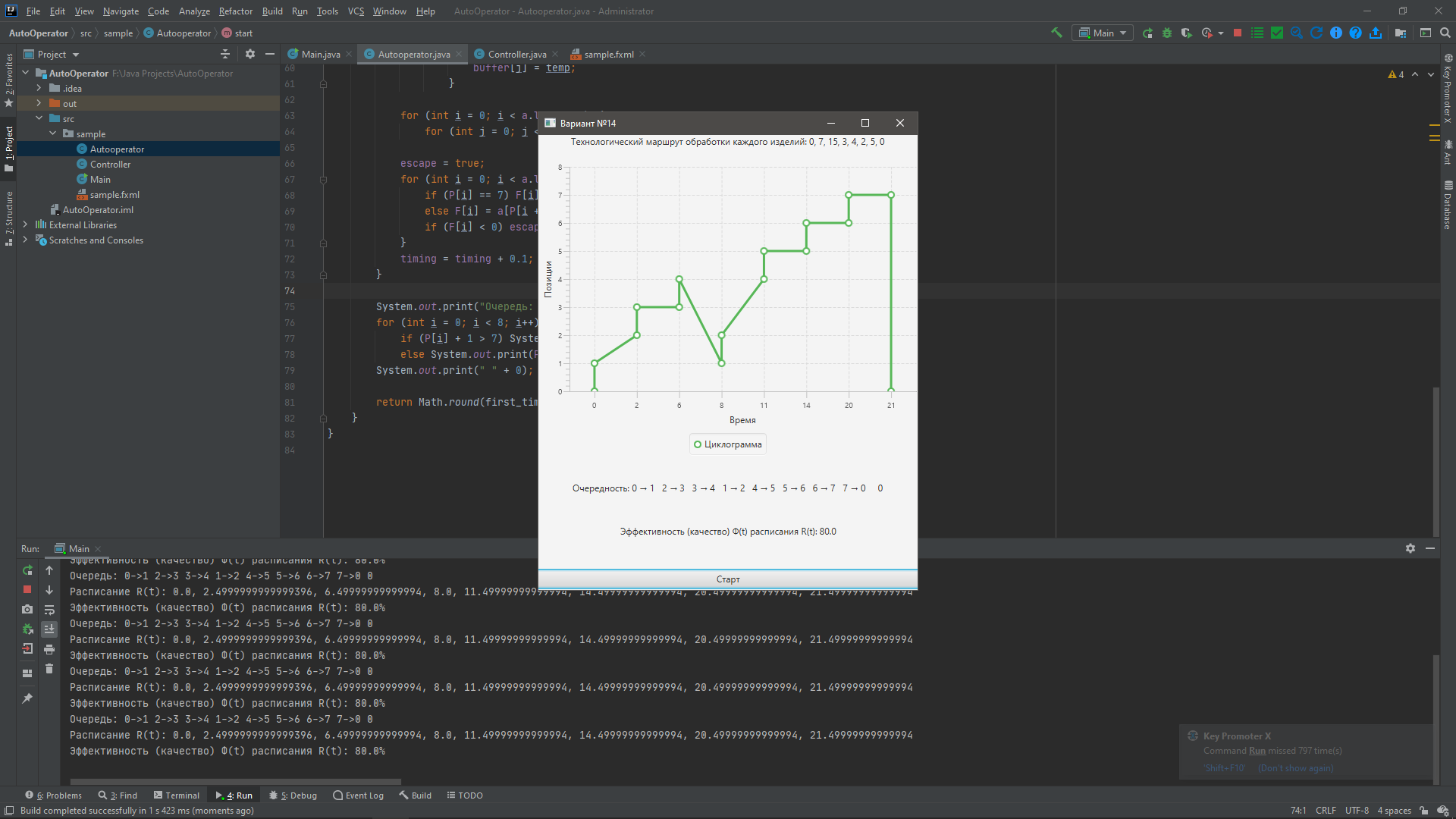
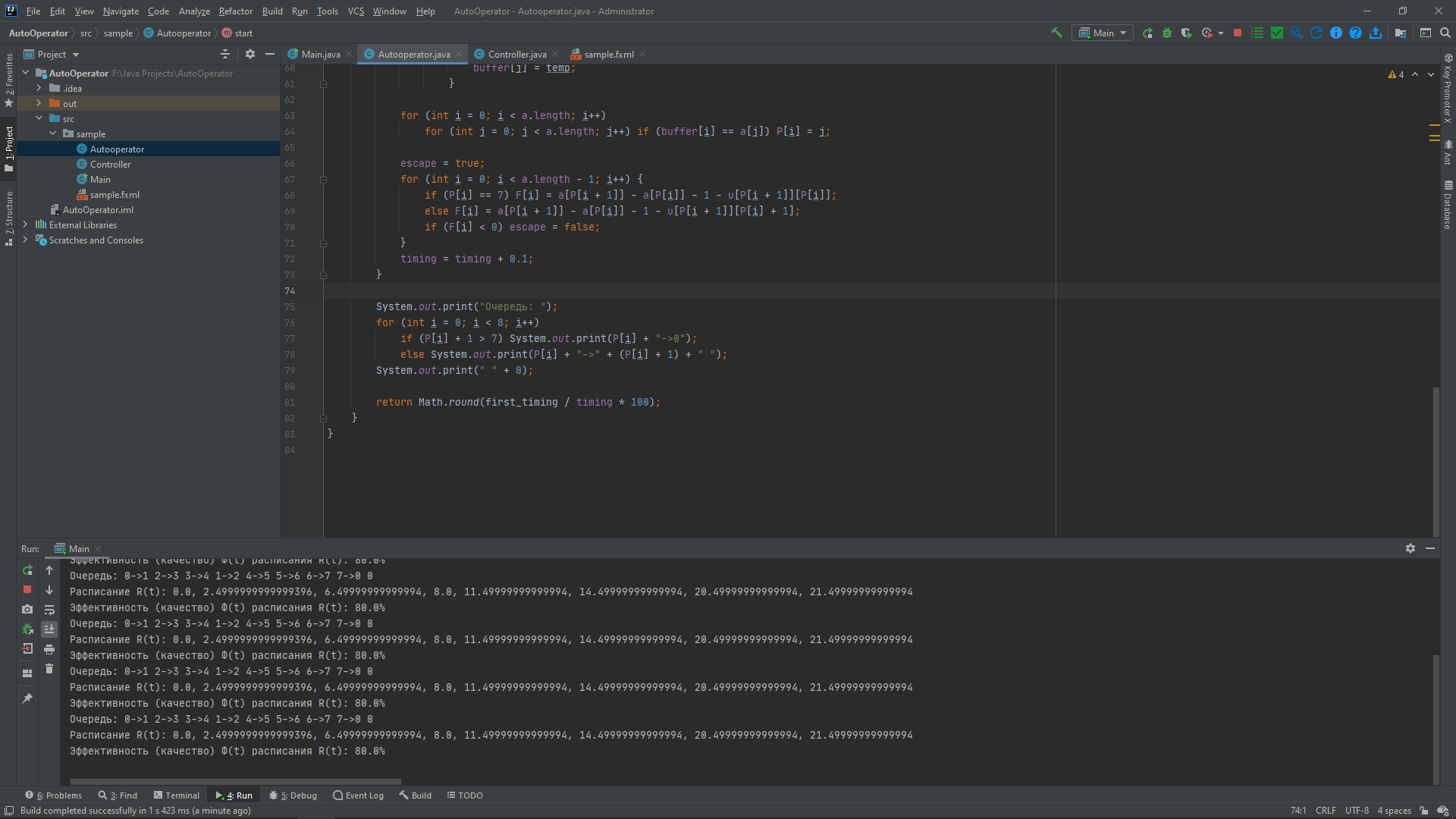
**ПРОТОТИП 1**

**КОД**

public class Autooperator {  
 public final double[] a = new double[7];  
 public final int[] P = new int[7];  
 public final double[] t;  
 private final double[][] u;  
 private final double[] A = new double[7];  
 private double[] F = new double[7];;  
 public double func;  
 private double[] buffer = new double[7];  
 private double[] levels = new double[7];  
 private boolean escape = false;  
 private double sum = 0;  
 private double timing = 0;  
  
 public Autooperator() {  
 t = new double[]{0, 7, 15, 3, 4, 2, 5};  
 u = new double[][]{  
 {0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6},  
 {0.1, 0, 0.1, 0.2, 0.3, 0.4, 0.5},  
 {0.2, 0.1, 0, 0.1, 0.2, 0.3, 0.4},  
 {0.3, 0.2, 0.1, 0, 0.1, 0.2, 0.3},  
 {0.4, 0.3, 0.2, 0.1, 0, 0.1, 0.2},  
 {0.5, 0.4, 0.3, 0.2, 0.1, 0, 0.1},  
 {0.6, 0.5, 0.4, 0.3, 0.2, 0.1, 0}  
 };  
 func = start();  
 System.*out*.println("\nРасписание R(t): " + a[P[0]] + ", " + a[P[1]] + ", " + a[P[2]] + ", " + a[P[3]] + ", " + a[P[4]] + ", " + a[P[5]] + ", " + a[P[6]]);  
 System.*out*.println("Эффективность (качество) Ф(t) расписания R(t): " + func + "%");  
 }  
  
 private double start() {  
 A[0] = 0;  
 for (int i = 1; i < 6; i++)  
 if ((t[i] + 1 + 1 + u[i - 1][i + 1]) > timing) timing = t[i] + 1 + 1 + u[i - 1][i + 1];  
 double first\_timing = timing;  
 for (int i = 0; i < 7; i++) sum = sum + t[i] + 1;  
 for (int i = 1; i < 7; i++) A[i] = A[i - 1] + t[i] + 1;  
  
 while (!escape) {  
 for (int i = 0; i < 7; i++) {  
 levels[i] = 0;  
 a[i] = A[i];  
 while (a[i] > timing) {  
 levels[i]++;  
 a[i] = A[i] - timing \* levels[i];  
 }  
 buffer[i] = a[i];  
 }  
  
 double temp;  
 for (int i = 0; i < a.length - 1; i++) {  
 for (int j = i + 1; j < a.length; j++) {  
 if (buffer[i] > buffer[j]) {  
 temp = buffer[i];  
 buffer[i] = buffer[j];  
 buffer[j] = temp;  
 }  
 }  
 }  
  
 for (int i = 0; i < a.length; i++)  
 for (int j = 0; j < a.length; j++) if (buffer[i] == a[j]) P[i] = j;  
  
 escape = true;  
 for (int i = 0; i < a.length - 1; i++) {  
 if (P[i] == 6) F[i] = a[P[i + 1]] - a[P[i]] - 1 - u[P[i + 1]][0];  
 else F[i] = a[P[i + 1]] - a[P[i]] - 1 - u[P[i + 1]][P[i] + 1];  
 if (F[i] < 0) escape = false;  
 }  
 timing = timing + 0.1;  
 }  
  
 System.*out*.print("\nОчередь: ");  
 for (int i = 0; i < 7; i++) {  
 if (P[i] + 1 > 6) System.*out*.print(P[i] + "->" + 0 + " ");  
 else System.*out*.print(P[i] + "->" + (P[i] + 1) + " ");  
 }  
 System.*out*.print(" " + 0 + "\n");  
  
 return Math.*round*((first\_timing / timing) \* 100);  
 }  
}

**ПРОТОТИП 2**

**КОД**

public Autooperator() {  
 t = new double[]{0, 7, 15, 3, 4, 2, 5, 0};  
 u = new double[][]{  
 {0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7},  
 {0.1, 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6},  
 {0.2, 0.1, 0, 0.1, 0.2, 0.3, 0.4, 0.5},  
 {0.3, 0.2, 0.1, 0, 0.1, 0.2, 0.3, 0.4},  
 {0.4, 0.3, 0.2, 0.1, 0, 0.1, 0.2, 0.3},  
 {0.5, 0.4, 0.3, 0.2, 0.1, 0, 0.1, 0.2},  
 {0.6, 0.5, 0.4, 0.3, 0.2, 0.1, 0, 0.1},  
 {0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1, 0}  
 };  
 func = start();  
 System.*out*.println("\nРасписание R(t): " + a[P[0]] + ", " + a[P[1]] + ", " + a[P[2]] + ", " + a[P[3]] + ", "  
 + a[P[4]] + ", " + a[P[5]] + ", " + a[P[6]] + ", " + a[P[7]]);  
 System.*out*.println("Эффективность (качество) Ф(t) расписания R(t): " + func + "%");  
}  
  
private double start() {  
 A[0] = 0;  
 for (int i = 1; i < 7; i++)  
 if (t[i] + 1 + 1 + u[i - 1][i + 1] > timing) timing = t[i] + 1 + 1 + u[i - 1][i + 1];  
 double first\_timing = timing;  
 for (int i = 0; i < 8; i++) sum = sum + t[i] + 1;  
 for (int i = 1; i < 8; i++) A[i] = A[i - 1] + t[i] + 1;  
  
 while (!escape) {  
 for (int i = 0; i < 8; i++) {  
 levels[i] = 0;  
 a[i] = A[i];  
 while (a[i] > timing) {  
 levels[i]++;  
 a[i] = A[i] - timing \* levels[i];  
 }  
 buffer[i] = a[i];  
 }  
  
 double temp;  
 for (int i = 0; i < a.length - 1; i++)  
 for (int j = i + 1; j < a.length; j++)  
 if (buffer[i] > buffer[j]) {  
 temp = buffer[i];  
 buffer[i] = buffer[j];  
 buffer[j] = temp;  
 }  
  
 for (int i = 0; i < a.length; i++)  
 for (int j = 0; j < a.length; j++) if (buffer[i] == a[j]) P[i] = j;  
  
 escape = true;  
 for (int i = 0; i < a.length - 1; i++) {  
 if (P[i] == 7) F[i] = a[P[i + 1]] - a[P[i]] - 1 - u[P[i + 1]][P[i]];  
 else F[i] = a[P[i + 1]] - a[P[i]] - 1 - u[P[i + 1]][P[i] + 1];  
 if (F[i] < 0) escape = false;  
 }  
 timing = timing + 0.1;  
 }  
  
 System.*out*.print("Очередь: ");  
 for (int i = 0; i < 8; i++)  
 if (P[i] + 1 > 7) System.*out*.print(P[i] + "->0");  
 else System.*out*.print(P[i] + "->" + (P[i] + 1) + " ");  
 System.*out*.print(" " + 0);  
  
 return Math.*round*(first\_timing / timing \* 100);  
}