2- Andrey Cheslow

```
package java44444;

import static java.lang.Math.abs;
import static java.lang.Math.pow;
import static java.lang.Math.sin;
import java.util.Random;

/**
 *
 * @author Andrey
 */
public class Java44444 {
   public static void main(String[] args) {
      int generation[][]=new int[20][22];
      int next_generation[][]=new int[20][22];

      Random rnd=new Random();
      //make first generation
      for(int i=0;i<20;i++){
         for(int j=0;j<22;j++){
           generation[i][j]=(rnd.nextInt(2));
         }
      }

      double fitness[] = new double[20];
      int pop=0;
      int num=0;
      while (num<1){
         System.out.println(pop+" population");
         pop++;
         if (fitness[0]>9)num ++;
         else num=0;
         for(int i=0;i<20;i++){
            fitness[i]=0;
         }

        for(int i=0;i<20;i++){ //calculating fitness
         int x1=0,x2=0;
       for(int k=0,f=10;k<10;k++,f--){
          x1+=(int) (generation[i][f]*pow(2,k));

          x2+=(int) (generation[i][f+11]*pow(2,k));

       }

       if(generation[i][0]==0) x1*=-1;
       if(generation[i][11]==0) x2*=-1;
       double x_1=((double)x1*5)/1023;
```

```
    double x_2=((double)x2*5)/1023;

        fitness[i]=x_1*sin(abs(x_1))+x_2*sin(abs(x_2));//71498314
      }
    double sum_fit=0;
      for(int i=0;i<20;i++){
        sum_fit+=fitness[i];//71498314
      }
      /*for(int i=0;i<20;i++){
        System.out.println(fitness[i]);
      }*/
      //sort generation according to ascending
      sort(fitness,generation);
      //calculating probability

      System.out.println(fitness[0]+" "+sum_fit/20);
      make_new_gen(generation,next_generation);
      /*for(int i=0;i<100;i++){
        System.out.println(number_of_choices[i]);
      } for(int i=0;i<20;i++){
        for(int j=0;j<22;j++){
          System.out.print(generation[i][j]/*+" "+sin(4));
        }
        System.out.println();
      }*/
      generation=next_generation;
    }
}
public static void sort(double[] fitness,int[][] generation){
    for(int i=0;i<19;i++){
      for(int j=i+1;j<20;j++){
        if (fitness[j]>fitness[i]){
          double temp=fitness[i];
          fitness[i]=fitness[j];
          fitness[j]=temp;
          int temps[]=generation[i];
            generation[i] = generation[j];
            generation[j] = temps;
        }
      }
    }
}
public static void make_new_gen(int[][] generation,int[][] next_generation){
    Random rnd= new Random();
    for(int i=0;i<20;i++){
      //truncate method
      int nump1=rnd.nextInt(10);
      int nump2=rnd.nextInt(10);
      //number_of_choices[nump1]++;
      //number_of_choices[nump2]++;
```

```
            for(int f=0;f<22;f++){
               int orrr=rnd.nextInt(2);
               //uniform crossover
               if(orrr==0)
               next_generation[i][f]=generation[nump1][f];
               else next_generation[i][f]=generation[nump2][f];
               }


         }
         int randomnumber=11;
            for(int k=0;k<20;k++){
            for(int j=0;j<22;j++){
              if(randomnumber==rnd.nextInt(21)){
                 if (next_generation[k][j]==0) next_generation[k][j]=1;
                 else next_generation[k][j]=0;
              }
            }
         }
      }
   }
```