Использование эволюционного подхода в задаче поиска глобального максимума гладкой функции

Постановка задачи.

Задана функция $f(x, y)$, она же функция приспособленности.

Генотип – значения аргументов (x, y).

Особенности задачи.

1. Быстрое вырождение популяции (можно обойтись популяцией малого объема):
2. Сильная зависимость от мутаций:
   - 2 особи – $[28\,000, 100\,000]_{\alpha=0,1} \rightarrow [2\,600, 19\,000]_{\alpha=0,5}$;
   - 10 особей – $[26\,000, 100\,000]_{\alpha=0,1} \rightarrow [5\,200, 18\,000]_{\alpha=0,5}$.
3. Наличие у целевой функции только одного экстремума.
4. Невозможность получения некорректного решения (нет ограничений).

Стратегия решения.

1. На каждой итерации выбираются 2 наиболее приспособленные особи A и B по критерию $f(x_i, y_i) \pm rd \rightarrow \min$, где $r \in [0,1]$ – случайное число с равномерным распределением, d – степень разброса (эмпирически выбрано $d = 0{,}1$).
2. Между ними проводится скрещивание с целью получения потомка C по принципу $(x_C, y_C) = \begin{cases} (x_A, y_A), r \le 0{,}25 \\ (x_A, y_B), 0{,}25 < r \le 0{,}5 \\ (x_B, y_A), 0{,}5 < r \le 0{,}75 \\ (x_B, y_B), 0{,}75 < r \le 1 \end{cases}$ – равновероятный выбор сочетания генов родителей.
3. Определяется, необходимо ли проводить мутацию по критерию $\alpha r_i > r_{i+1}$, где α – вероятность мутации. Если мутация выполняется, то случайная координата (x или y с вероятностью 0,5) заменяется случайным значением.
4. Потомок замещает наименее приспособленную особь популяции по критерию $f(x_i, y_i) \pm rd \rightarrow \max$.
5. Итерации повторяются до тех пор, пока вся популяция не соберется близко от глобального экстремума и $\left| f - f^* \right| < 1 \cdot 10^{-5}$.

Листинг программы.

```
program ga_func;
{$APPTYPE CONSOLE}

uses
  SysUtils;

const
  POPULATION_SIZE = 10;             { Размер популяции }
  MUTATION_PROBABILITY = 0.5;       { Вероятность мутации }

type
  TGenotype = record
    X, Y: Double;
  end;

var
  PopulationArr: array [1..POPULATION_SIZE] of TGenotype;

{ Случайная илициализация популяции (-2; 2) }
```

```
procedure RandomInit();
var
  I: Integer;
begin
  for I := 1 to POPULATION_SIZE do begin
    PopulationArr[I].X := Random*4 - 2;
    PopulationArr[I].Y := Random*4 - 2;
  end;
end;


{ Функция приспособленности (целевая оптимизируемая функция) }
function Fitness(Index: Integer): Double;
begin
  Result := Sqr(PopulationArr[Index].X) + Sqr(PopulationArr[Index].Y);
end;


{ Расчет средней приспособленности популяции }
function GetAverageFitness(): Double;
var
  I: Integer;
begin
  Result := 0.0;

  for I := 1 to POPULATION_SIZE do
    Result := Result + Fitness(I);

  Result := Result / POPULATION_SIZE;
end;


{ Вывод популяции на экран }
procedure WritePopulation();
var
  I: Integer;
begin
  for I := 1 to POPULATION_SIZE do
    Writeln(I:3, ' x=', PopulationArr[I].X:6:3, ' y=', PopulationArr[I].Y:6:3, ' (',
        Fitness(I):5:3, ')');

  Writeln('Average fitness = ', GetAverageFitness():5:3);
end;


{ Скрещивание }
function Crossover(MotherIndex, FatherIndex: Integer): TGenotype;
begin
  if Random > 0.5 then
    Result.X := PopulationArr[MotherIndex].X
  else
    Result.X := PopulationArr[FatherIndex].X;

  if Random > 0.5 then
    Result.Y := PopulationArr[MotherIndex].Y
  else
    Result.Y := PopulationArr[FatherIndex].Y;
end;


{ Мутация }
procedure Mutation(var Son: TGenotype);
var
  NewValue: Double;
begin
  { Новое значение координаты }
  NewValue := Random*4 - 2;

  { Определение координаты, которая подлежит мутации }
  if Random > 0.5 then
    Son.X := NewValue
  else
    Son.Y := NewValue;
end;


{ Расчет приспособленности с отклонением }
function FitnessWithDeviation(SourceFitness: Double): Double;
const
  DEVIATION = 0.1;
```

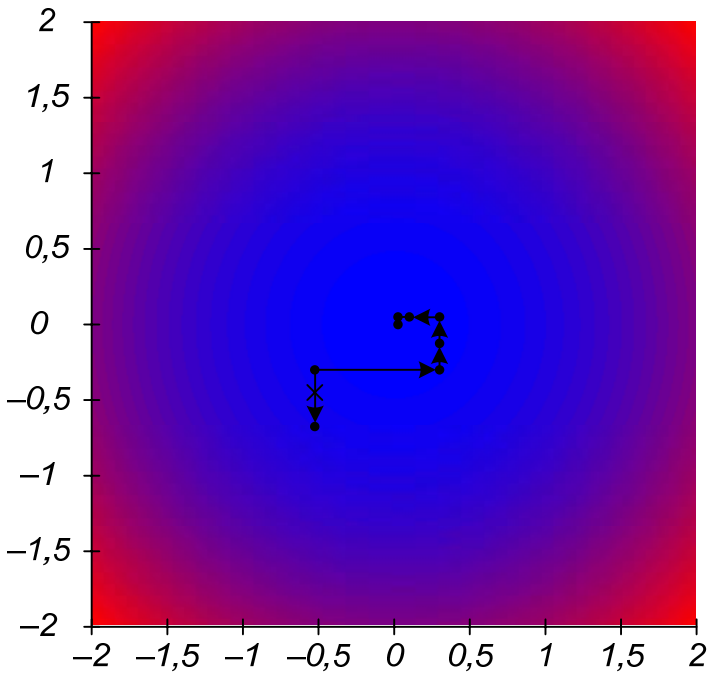
```
begin
  Result := SourceFitness * (1 + Random*DEVIATION - DEVIATION/2);
end;


{ Выбор наиболее приспособленного родителя }
function GetBest(SkipThis: Integer = -1): Integer;
var
  BestFitness, CurrFitness: Double;
  BestIndex, I: Integer;
begin
  BestFitness := FitnessWithDeviation( Fitness(1) );
  BestIndex := 1;

  for I := 2 to POPULATION_SIZE do begin
    if I = SkipThis then
      continue;

    CurrFitness := FitnessWithDeviation( Fitness(I) );

    if CurrFitness < BestFitness then begin
      BestFitness := CurrFitness;
      BestIndex := I;
    end;
  end;

  Result := BestIndex;
end;


{ Выбор наименее приспособленной особи }
function GetWorst(): Integer;
var
  WorstFitness, CurrFitness: Double;
  WorstIndex, I: Integer;
begin
  WorstFitness := FitnessWithDeviation( Fitness(1) );
  WorstIndex := 1;

  for I := 2 to POPULATION_SIZE do begin
    CurrFitness := FitnessWithDeviation( Fitness(I) );

    if CurrFitness > WorstFitness then begin
      WorstFitness := CurrFitness;
      WorstIndex := I;
    end;
  end;

  Result := WorstIndex;
end;


var
  Iteration, MotherIndex, FatherIndex, DeadIndex: Integer;
  S: TGenotype;

begin
  RandSeed := 4;
  RandomInit();

  Iteration := 0;

  repeat
    Writeln('Iteration = ', Iteration);
    WritePopulation();
    Writeln;

    { Выбор особей для скрещивания }
    MotherIndex := GetBest();
    FatherIndex := GetBest(MotherIndex);

    { Скрещивание }
    S := Crossover(MotherIndex, FatherIndex);

    { Мутация }
    if Random*MUTATION_PROBABILITY > Random then
      Mutation(S);

    { Выбор наиболее слабой особи }
    DeadIndex := GetWorst();
```

```
    { Новая особь заменяет наименее приспособленную }
    PopulationArr[DeadIndex] := S;

    Inc(Iteration);

    { Расчет средней приспособленности }
    if (Abs(GetAverageFitness()) < 4e-6) or (Iteration > 100000) then
      break;
  until False;

  { Вывод итоговой популяции }
  Writeln('Iteration = ', Iteration);
  WritePopulation();
  Writeln;

  Writeln('Done');
  Readln;
end.
```

Пример 1 – функция с одним экстремумом.

$f(x, y) = x^2 + y^2 \to \min$, $-2 \le x \le 2, -2 \le y \le 2$.

Глобальный минимум: $f^* = 0$ при $x^* = 0$ и $y^* = 0$.



Протокол эволюции.

```
Iteration = 0                          ; Номер итерации при моделировании
  1 x=-2.000 y=-1.874 (7.514)          ; Номер_особи Фенотип(x, y) (Приспособленность)
  2 x= 1.444 y=-1.190 (3.501)
  3 x=-0.908 y= 0.687 (1.296)
  4 x=-0.725 y=-1.353 (2.356)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-1.672 y=-0.101 (2.806)
  7 x=-1.718 y= 1.363 (4.809)
  8 x=-1.761 y=-0.827 (3.785)
  9 x= 1.669 y=-0.528 (3.065)
 10 x= 1.099 y=-0.688 (1.681)
Average fitness = 3.116

Iteration = 1
  1 x=-0.908 y= 0.687 (1.296)
  2 x= 1.444 y=-1.190 (3.501)
  3 x=-0.908 y= 0.687 (1.296)
  4 x=-0.725 y=-1.353 (2.356)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-1.672 y=-0.101 (2.806)
  7 x=-1.718 y= 1.363 (4.809)
  8 x=-1.761 y=-0.827 (3.785)
  9 x= 1.669 y=-0.528 (3.065)
 10 x= 1.099 y=-0.688 (1.681)
Average fitness = 2.495
```

```
Iteration = 2
  1 x=-0.908 y= 0.687 (1.296)
  2 x= 1.444 y=-1.190 (3.501)
  3 x=-0.908 y= 0.687 (1.296)
  4 x=-0.725 y=-1.353 (2.356)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-1.672 y=-0.101 (2.806)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-1.761 y=-0.827 (3.785)
  9 x= 1.669 y=-0.528 (3.065)
 10 x= 1.099 y=-0.688 (1.681)
Average fitness = 2.049

Iteration = 3
  1 x=-0.908 y= 0.687 (1.296)
  2 x= 1.444 y=-1.190 (3.501)
  3 x=-0.908 y= 0.687 (1.296)
  4 x=-0.725 y=-1.353 (2.356)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-1.672 y=-0.101 (2.806)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x= 1.669 y=-0.528 (3.065)
 10 x= 1.099 y=-0.688 (1.681)
Average fitness = 1.705

Iteration = 4
  1 x=-0.908 y= 0.687 (1.296)
  2 x=-0.511 y=-0.297 (0.350)
  3 x=-0.908 y= 0.687 (1.296)
  4 x=-0.725 y=-1.353 (2.356)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-1.672 y=-0.101 (2.806)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x= 1.669 y=-0.528 (3.065)
 10 x= 1.099 y=-0.688 (1.681)
Average fitness = 1.390

Iteration = 5
  1 x=-0.908 y= 0.687 (1.296)
  2 x=-0.511 y=-0.297 (0.350)
  3 x=-0.908 y= 0.687 (1.296)
  4 x=-0.725 y=-1.353 (2.356)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-1.672 y=-0.101 (2.806)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x=-0.761 y=-0.297 (0.667)
 10 x= 1.099 y=-0.688 (1.681)
Average fitness = 1.150

Iteration = 6
  1 x=-0.908 y= 0.687 (1.296)
  2 x=-0.511 y=-0.297 (0.350)
  3 x=-0.908 y= 0.687 (1.296)
  4 x=-0.725 y=-1.353 (2.356)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-0.511 y=-0.297 (0.350)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x=-0.761 y=-0.297 (0.667)
 10 x= 1.099 y=-0.688 (1.681)
Average fitness = 0.904

Iteration = 7
  1 x=-0.908 y= 0.687 (1.296)
  2 x=-0.511 y=-0.297 (0.350)
  3 x=-0.908 y= 0.687 (1.296)
  4 x=-0.511 y=-0.297 (0.350)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-0.511 y=-0.297 (0.350)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x=-0.761 y=-0.297 (0.667)
 10 x= 1.099 y=-0.688 (1.681)
Average fitness = 0.704

Iteration = 8
  1 x=-0.908 y= 0.687 (1.296)
  2 x=-0.511 y=-0.297 (0.350)
  3 x=-0.908 y= 0.687 (1.296)
  4 x=-0.511 y=-0.297 (0.350)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-0.511 y=-0.297 (0.350)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x=-0.761 y=-0.297 (0.667)
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.571

Iteration = 9
  1 x=-0.908 y= 0.687 (1.296)
  2 x=-0.511 y=-0.297 (0.350)
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-0.511 y=-0.297 (0.350)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
```

```
  9 x=-0.761 y=-0.297 (0.667)
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.476

Iteration = 10
  1 x=-0.511 y=-0.297 (0.350)
  2 x=-0.511 y=-0.297 (0.350)
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-0.511 y=-0.297 (0.350)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x=-0.761 y=-0.297 (0.667)
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.381

Iteration = 11
  1 x=-0.511 y=-0.297 (0.350)
  2 x=-0.511 y=-0.297 (0.350)
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-0.511 y=-0.297 (0.350)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x=-0.511 y=-0.297 (0.350)
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.350
```

Вырождение популяции

```
[skip]

Iteration = 22
  1 x=-0.511 y=-0.297 (0.350)
  2 x=-0.511 y=-0.297 (0.350)
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x=-0.511 y=-0.297 (0.350)
  6 x= 0.672 y=-0.297 (0.540)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x=-0.511 y=-0.297 (0.350)
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.369

Iteration = 23
  1 x=-0.511 y=-0.297 (0.350)
  2 x=-0.511 y=-0.297 (0.350)
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-0.511 y=-0.297 (0.350)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x=-0.511 y=-0.297 (0.350)
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.350

[skip]
```

; Первая мутация, неудачная

Первая удачная мутация, популяция начинает переползать в более приспособленный максимум

```
Iteration = 444
  1 x=-0.511 y=-0.297 (0.350)
  2 x= 0.301 y=-0.297 (0.179)
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-0.511 y=-0.297 (0.350)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x=-0.511 y=-0.297 (0.350)
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.332

Iteration = 445
  1 x=-0.511 y=-0.297 (0.350)
  2 x= 0.301 y=-0.297 (0.179)
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-0.511 y=-0.297 (0.350)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x=-0.511 y=-0.297 (0.350)
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.332

Iteration = 446
  1 x=-0.511 y=-0.297 (0.350)
  2 x= 0.301 y=-0.297 (0.179)
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-0.511 y=-0.297 (0.350)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x=-0.511 y=-0.297 (0.350)
 10 x=-0.511 y=-0.297 (0.350)
```

; !!!

```
Average fitness = 0.332

Iteration = 447
  1 x=-0.511 y=-0.297 (0.350)
  2 x= 0.301 y=-0.297 (0.179)
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-0.511 y=-0.297 (0.350)
  7 x=-0.511 y=-0.297 (0.350)
  8 x=-0.511 y=-0.297 (0.350)
  9 x=-0.511 y=-0.297 (0.350)
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.332

Iteration = 448
  1 x=-0.511 y=-0.297 (0.350)
  2 x= 0.301 y=-0.297 (0.179)          ; !!!
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x=-0.511 y=-0.297 (0.350)
  6 x=-0.511 y=-0.297 (0.350)
  7 x=-0.511 y=-0.297 (0.350)
  8 x= 0.301 y=-0.297 (0.179)          ; !!!
  9 x=-0.511 y=-0.297 (0.350)
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.315

Iteration = 449
  1 x=-0.511 y=-0.297 (0.350)
  2 x= 0.301 y=-0.297 (0.179)          ; !!!
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x=-0.511 y=-0.297 (0.350)
  6 x= 0.301 y=-0.297 (0.179)          ; !!!
  7 x=-0.511 y=-0.297 (0.350)
  8 x= 0.301 y=-0.297 (0.179)          ; !!!
  9 x=-0.511 y=-0.297 (0.350)
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.298

Iteration = 450
  1 x=-0.511 y=-0.297 (0.350)
  2 x= 0.301 y=-0.297 (0.179)
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x= 0.381 y=-0.297 (0.234)
  6 x= 0.301 y=-0.297 (0.179)
  7 x=-0.511 y=-0.297 (0.350)
  8 x= 0.301 y=-0.297 (0.179)
  9 x=-0.511 y=-0.297 (0.350)
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.287

Iteration = 451
  1 x=-0.511 y=-0.297 (0.350)
  2 x= 0.301 y=-0.297 (0.179)          ; !!!
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x= 0.381 y=-0.297 (0.234)
  6 x= 0.301 y=-0.297 (0.179)          ; !!!
  7 x=-0.511 y=-0.297 (0.350)
  8 x= 0.301 y=-0.297 (0.179)          ; !!!
  9 x= 0.301 y=-0.297 (0.179)          ; !!!
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.270

Iteration = 452
  1 x=-0.511 y=-0.297 (0.350)
  2 x= 0.301 y=-0.297 (0.179)          ; !!!
  3 x=-0.511 y=-0.297 (0.350)
  4 x=-0.511 y=-0.297 (0.350)
  5 x= 0.381 y=-0.297 (0.234)
  6 x= 0.301 y=-0.297 (0.179)          ; !!!
  7 x= 0.301 y=-0.297 (0.179)          ; !!!
  8 x= 0.301 y=-0.297 (0.179)          ; !!!
  9 x= 0.301 y=-0.297 (0.179)          ; !!!
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.253

Iteration = 453
  1 x=-0.511 y=-0.297 (0.350)
  2 x= 0.301 y=-0.297 (0.179)          ; !!!
  3 x=-0.511 y=-0.297 (0.350)
  4 x= 0.301 y=-0.297 (0.179)          ; !!!
  5 x= 0.381 y=-0.297 (0.234)
  6 x= 0.301 y=-0.297 (0.179)          ; !!!
  7 x= 0.301 y=-0.297 (0.179)          ; !!!
  8 x= 0.301 y=-0.297 (0.179)          ; !!!
  9 x= 0.301 y=-0.297 (0.179)          ; !!!
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.236

Iteration = 454
  1 x=-0.511 y=-0.297 (0.350)
  2 x= 0.301 y=-0.297 (0.179)          ; !!!
  3 x= 0.301 y=-0.297 (0.179)          ; !!!
  4 x= 0.301 y=-0.297 (0.179)          ; !!!
  5 x= 0.381 y=-0.297 (0.234)
  6 x= 0.301 y=-0.297 (0.179)          ; !!!
```

```
  7 x= 0.301 y=-0.297 (0.179)          ; !!!
  8 x= 0.301 y=-0.297 (0.179)          ; !!!
  9 x= 0.301 y=-0.297 (0.179)          ; !!!
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.219

Iteration = 455
  1 x= 0.301 y=-0.297 (0.179)          ; !!!
  2 x= 0.301 y=-0.297 (0.179)          ; !!!
  3 x= 0.301 y=-0.297 (0.179)          ; !!!
  4 x= 0.301 y=-0.297 (0.179)          ; !!!
  5 x= 0.381 y=-0.297 (0.234)
  6 x= 0.301 y=-0.297 (0.179)          ; !!!
  7 x= 0.301 y=-0.297 (0.179)          ; !!!
  8 x= 0.301 y=-0.297 (0.179)          ; !!!
  9 x= 0.301 y=-0.297 (0.179)          ; !!!
 10 x=-0.511 y=-0.297 (0.350)
Average fitness = 0.201

Iteration = 456
  1 x= 0.301 y=-0.297 (0.179)          ; !!!
  2 x= 0.301 y=-0.297 (0.179)          ; !!!
  3 x= 0.301 y=-0.297 (0.179)          ; !!!
  4 x= 0.301 y=-0.297 (0.179)          ; !!!
  5 x= 0.381 y=-0.297 (0.234)
  6 x= 0.301 y=-0.297 (0.179)          ; !!!
  7 x= 0.301 y=-0.297 (0.179)          ; !!!
  8 x= 0.301 y=-0.297 (0.179)          ; !!!
  9 x= 0.301 y=-0.297 (0.179)          ; !!!
 10 x= 0.301 y=-0.297 (0.179)          ; !!!
Average fitness = 0.184
```

Популяция полностью переползла в новый локальный максимум и выродилась

```
Iteration = 457
  1 x= 0.301 y=-0.297 (0.179)
  2 x= 0.301 y=-0.297 (0.179)
  3 x= 0.301 y=-0.297 (0.179)
  4 x= 0.301 y=-0.297 (0.179)
  5 x= 0.301 y=-0.297 (0.179)
  6 x= 0.301 y=-0.297 (0.179)
  7 x= 0.301 y=-0.297 (0.179)
  8 x= 0.301 y=-0.297 (0.179)
  9 x= 0.301 y=-0.297 (0.179)
 10 x= 0.301 y=-0.297 (0.179)
Average fitness = 0.179

[skip]

Iteration = 525
  1 x= 0.301 y=-0.297 (0.179)
  2 x= 0.301 y=-0.132 (0.108)          ; !!!
  3 x= 0.301 y=-0.297 (0.179)
  4 x= 0.301 y=-0.297 (0.179)
  5 x= 0.301 y=-0.297 (0.179)
  6 x= 0.301 y=-0.297 (0.179)
  7 x= 0.301 y=-0.297 (0.179)
  8 x= 0.301 y=-0.297 (0.179)
  9 x= 0.301 y=-0.297 (0.179)
 10 x= 0.301 y=-0.297 (0.179)
Average fitness = 0.172

[skip]

Iteration = 859
  1 x= 0.301 y=-0.132 (0.108)
  2 x= 0.301 y=-0.132 (0.108)
  3 x= 0.301 y=-0.132 (0.108)
  4 x= 0.301 y=-0.132 (0.108)
  5 x= 0.301 y=-0.132 (0.108)
  6 x= 0.301 y=-0.132 (0.108)
  7 x= 0.301 y=-0.132 (0.108)
  8 x= 0.301 y=-0.132 (0.108)
  9 x= 0.301 y= 0.107 (0.102)          ; !!!
 10 x= 0.301 y=-0.132 (0.108)
Average fitness = 0.107

[skip]

Iteration = 871
  1 x= 0.301 y= 0.107 (0.102)
  2 x= 0.301 y= 0.041 (0.092)          ; !!!
  3 x= 0.301 y= 0.107 (0.102)
  4 x= 0.301 y= 0.107 (0.102)
  5 x= 0.301 y= 0.107 (0.102)
  6 x= 0.301 y= 0.107 (0.102)
  7 x= 0.301 y= 0.107 (0.102)
  8 x= 0.301 y= 0.107 (0.102)
  9 x= 0.301 y= 0.107 (0.102)
 10 x= 0.301 y= 0.107 (0.102)
Average fitness = 0.101

[skip]

Iteration = 1112
  1 x= 0.301 y= 0.041 (0.092)
  2 x= 0.106 y= 0.041 (0.013)          ; !!!
  3 x= 0.301 y= 0.041 (0.092)
  4 x= 0.301 y= 0.041 (0.092)
  5 x= 0.301 y= 0.041 (0.092)
```

```
  6 x= 0.301 y= 0.041 (0.092)
  7 x= 0.301 y= 0.041 (0.092)
  8 x= 0.301 y= 0.041 (0.092)
  9 x= 0.301 y= 0.041 (0.092)
 10 x= 0.301 y= 0.041 (0.092)
Average fitness = 0.084

[skip]

Iteration = 1457
  1 x= 0.106 y= 0.041 (0.013)
  2 x= 0.106 y= 0.041 (0.013)
  3 x= 0.106 y= 0.041 (0.013)
  4 x= 0.106 y= 0.041 (0.013)
  5 x= 0.106 y= 0.041 (0.013)
  6 x= 0.106 y= 0.041 (0.013)
  7 x= 0.025 y= 0.041 (0.002)          ; !!!
  8 x= 0.106 y= 0.041 (0.013)
  9 x= 0.106 y= 0.041 (0.013)
 10 x= 0.106 y= 0.041 (0.013)
Average fitness = 0.012

[skip]

Iteration = 2701
  1 x= 0.016 y= 0.041 (0.002)
  2 x= 0.016 y= 0.041 (0.002)
  3 x= 0.016 y= 0.041 (0.002)
  4 x= 0.016 y= 0.041 (0.002)
  5 x= 0.016 y= 0.041 (0.002)
  6 x= 0.016 y= 0.041 (0.002)
  7 x= 0.016 y=-0.032 (0.001)          ; !!!
  8 x= 0.016 y= 0.041 (0.002)
  9 x= 0.016 y= 0.041 (0.002)
 10 x= 0.016 y= 0.041 (0.002)
Average fitness = 0.002

[skip]

Iteration = 2884
  1 x= 0.016 y=-0.032 (0.001)
  2 x= 0.016 y=-0.032 (0.001)
  3 x= 0.016 y=-0.032 (0.001)
  4 x= 0.016 y=-0.032 (0.001)
  5 x= 0.016 y=-0.032 (0.001)
  6 x= 0.016 y=-0.032 (0.001)
  7 x= 0.016 y=-0.032 (0.001)
  8 x= 0.016 y=-0.032 (0.001)
  9 x= 0.016 y=-0.032 (0.001)
 10 x=-1.497 y=-0.032 (2.242)         ; Время от времени появляются неудачные мутации
Average fitness = 0.225

[skip]

Iteration = 4524
  1 x= 0.016 y=-0.032 (0.001)
  2 x= 0.016 y=-0.032 (0.001)
  3 x= 0.016 y=-0.032 (0.001)
  4 x= 0.016 y=-0.032 (0.001)
  5 x= 0.016 y=-0.032 (0.001)
  6 x= 0.016 y= 0.009 (0.000)          ; !!!
  7 x= 0.016 y=-0.032 (0.001)
  8 x= 0.016 y=-0.032 (0.001)
  9 x= 0.016 y=-0.032 (0.001)
 10 x= 0.016 y=-0.032 (0.001)
Average fitness = 0.001

[skip]

Iteration = 4631
  1 x= 0.016 y= 0.009 (0.000)
  2 x= 0.016 y= 0.009 (0.000)
  3 x= 0.015 y= 0.009 (0.000)          ; некоторые мутации совсем незначительны
  4 x= 0.016 y= 0.009 (0.000)
  5 x= 0.016 y= 0.009 (0.000)
  6 x= 0.016 y= 0.009 (0.000)
  7 x= 0.016 y= 0.009 (0.000)
  8 x= 0.016 y= 0.009 (0.000)
  9 x= 0.016 y= 0.009 (0.000)
 10 x= 0.016 y= 0.009 (0.000)
Average fitness = 0.000

[skip]

Iteration = 5316
  1 x= 0.015 y= 0.009 (0.000)
  2 x=-0.006 y= 0.009 (0.000)          ; !!!
  3 x= 0.015 y= 0.009 (0.000)
  4 x= 0.015 y= 0.009 (0.000)
  5 x= 0.015 y= 0.009 (0.000)
  6 x= 0.015 y= 0.009 (0.000)
  7 x= 0.015 y= 0.009 (0.000)
  8 x= 0.015 y= 0.009 (0.000)
  9 x= 0.015 y= 0.009 (0.000)
 10 x= 0.015 y= 0.009 (0.000)
Average fitness = 0.000

[skip]

Iteration = 11597
```

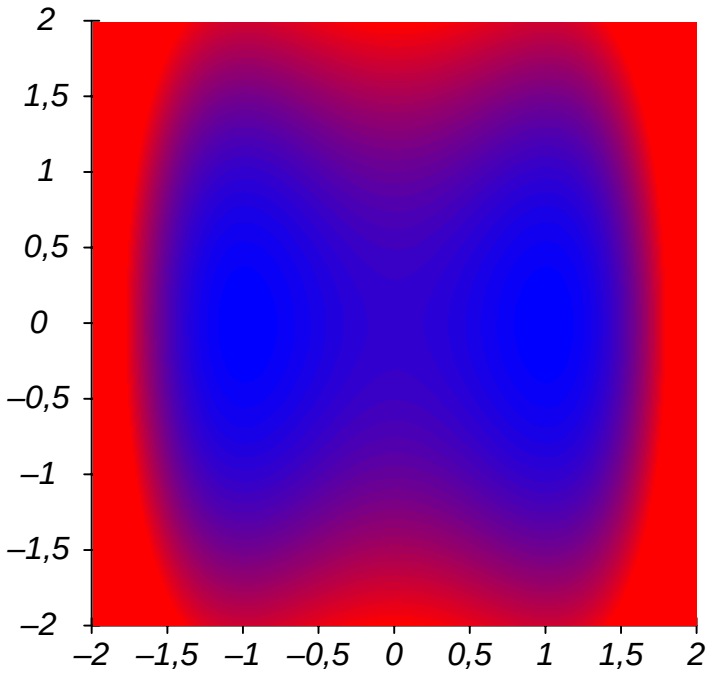
```
  1 x=-0.006 y= 0.009 (0.000)
  2 x=-0.006 y= 0.009 (0.000)
  3 x=-0.006 y= 0.009 (0.000)
  4 x=-0.006 y= 0.009 (0.000)
  5 x=-0.006 y= 0.009 (0.000)
  6 x=-0.006 y= 0.009 (0.000)
  7 x=-0.006 y=-0.001 (0.000)          ; !!!
  8 x=-0.006 y= 0.009 (0.000)
  9 x=-0.006 y= 0.009 (0.000)
 10 x=-0.006 y= 0.009 (0.000)
Average fitness = 0.000

[skip]

Iteration = 18896
  1 x=-0.006 y=-0.001 (0.000)
  2 x=-0.006 y=-0.001 (0.000)
  3 x=-0.006 y=-0.001 (0.000)
  4 x= 0.003 y=-0.001 (0.000)          ; !!!
  5 x=-0.006 y=-0.001 (0.000)
  6 x=-0.006 y=-0.001 (0.000)
  7 x=-0.006 y=-0.001 (0.000)
  8 x=-0.006 y=-0.001 (0.000)
  9 x=-0.006 y=-0.001 (0.000)
 10 x=-0.006 y=-0.001 (0.000)
Average fitness = 0.000

[skip]

Iteration = 27468
  1 x= 0.003 y=-0.001 (0.000)
  2 x= 0.003 y=-0.001 (0.000)
  3 x= 0.003 y=-0.001 (0.000)
  4 x= 0.003 y=-0.001 (0.000)
  5 x=-0.001 y=-0.001 (0.000)          ; !!!
  6 x= 0.003 y=-0.001 (0.000)
  7 x= 0.003 y=-0.001 (0.000)
  8 x= 0.003 y=-0.001 (0.000)
  9 x= 0.003 y=-0.001 (0.000)
 10 x= 0.003 y=-0.001 (0.000)
Average fitness = 0.000
```

Далее до 100000 итераций улучшения приспособленности популяции не происходит: лучшее решение x=-0.001, y=-0.001

Пример 2 – функция с двумя экстремумами.

$f(x, y) = (x-1)^2 (x+1)^2 + y^2 \to \min,\ -2 \le x \le 2, -2 \le y \le 2.$

Глобальный минимум: $f^* = 0$ при $x^* = -1$ и $y^* = 0$ или $x^* = +1$ и $y^* = 0$.



Протокол эволюции.

```
Iteration = 0
  1 x=-2.000 y=-1.874 (12.514)
  2 x= 1.444 y=-1.190 (2.594)
  3 x=-0.908 y= 0.687 (0.502)
  4 x=-0.725 y=-1.353 (2.055)
  5 x=-0.511 y=-0.297 (0.634)
  6 x=-1.672 y=-0.101 (3.234)
```

```
  7 x=-1.718 y= 1.363 (5.664)
  8 x=-1.761 y=-0.827 (5.100)
  9 x= 1.669 y=-0.528 (3.469)
 10 x= 1.099 y=-0.688 (0.517)
Average fitness = 3.628

Iteration = 1
  1 x=-0.908 y= 0.687 (0.502)
  2 x= 1.444 y=-1.190 (2.594)
  3 x=-0.908 y= 0.687 (0.502)
  4 x=-0.725 y=-1.353 (2.055)
  5 x=-0.511 y=-0.297 (0.634)
  6 x=-1.672 y=-0.101 (3.234)
  7 x=-1.718 y= 1.363 (5.664)
  8 x=-1.761 y=-0.827 (5.100)
  9 x= 1.669 y=-0.528 (3.469)
 10 x= 1.099 y=-0.688 (0.517)
Average fitness = 2.427

Iteration = 2
  1 x=-0.908 y= 0.687 (0.502)
  2 x= 1.444 y=-1.190 (2.594)
  3 x=-0.908 y= 0.687 (0.502)
  4 x=-0.725 y=-1.353 (2.055)
  5 x=-0.511 y=-0.297 (0.634)
  6 x=-1.672 y=-0.101 (3.234)
  7 x=-0.908 y= 0.687 (0.502)
  8 x=-1.761 y=-0.827 (5.100)
  9 x= 1.669 y=-0.528 (3.469)
 10 x= 1.099 y=-0.688 (0.517)
Average fitness = 1.911

Iteration = 3
  1 x=-0.908 y= 0.687 (0.502)
  2 x= 1.444 y=-1.190 (2.594)
  3 x=-0.908 y= 0.687 (0.502)
  4 x=-0.725 y=-1.353 (2.055)
  5 x=-0.511 y=-0.297 (0.634)
  6 x=-1.672 y=-0.101 (3.234)
  7 x=-0.908 y= 0.687 (0.502)
  8 x=-0.908 y= 0.687 (0.502)
  9 x= 1.669 y=-0.528 (3.469)
 10 x= 1.099 y=-0.688 (0.517)
Average fitness = 1.451

Iteration = 4
  1 x=-0.908 y= 0.687 (0.502)
  2 x= 1.444 y=-1.190 (2.594)
  3 x=-0.908 y= 0.687 (0.502)
  4 x=-0.725 y=-1.353 (2.055)
  5 x=-0.511 y=-0.297 (0.634)
  6 x=-1.672 y=-0.101 (3.234)
  7 x=-0.908 y= 0.687 (0.502)
  8 x=-0.908 y= 0.687 (0.502)
  9 x=-0.908 y= 0.687 (0.502)
 10 x= 1.099 y=-0.688 (0.517)
Average fitness = 1.154

Iteration = 5
  1 x=-0.908 y= 0.687 (0.502)
  2 x= 1.444 y=-1.190 (2.594)
  3 x=-0.908 y= 0.687 (0.502)
  4 x=-0.725 y=-1.353 (2.055)
  5 x=-0.511 y=-0.297 (0.634)
  6 x=-0.908 y= 0.687 (0.502)
  7 x=-0.908 y= 0.687 (0.502)
  8 x=-0.908 y= 0.687 (0.502)
  9 x=-0.908 y= 0.687 (0.502)
 10 x= 1.099 y=-0.688 (0.517)
Average fitness = 0.881

Iteration = 6
  1 x=-0.908 y= 0.687 (0.502)
  2 x=-0.908 y= 0.687 (0.502)
  3 x=-0.908 y= 0.687 (0.502)
  4 x=-0.725 y=-1.353 (2.055)
  5 x=-0.511 y=-0.297 (0.634)
  6 x=-0.908 y= 0.687 (0.502)
  7 x=-0.908 y= 0.687 (0.502)
  8 x=-0.908 y= 0.687 (0.502)
  9 x=-0.908 y= 0.687 (0.502)
 10 x= 1.099 y=-0.688 (0.517)
Average fitness = 0.672

Iteration = 7
  1 x=-0.908 y= 0.687 (0.502)
  2 x=-0.908 y= 0.687 (0.502)
  3 x=-0.908 y= 0.687 (0.502)
  4 x=-0.908 y= 0.687 (0.502)
  5 x=-0.511 y=-0.297 (0.634)
  6 x=-0.908 y= 0.687 (0.502)
  7 x=-0.908 y= 0.687 (0.502)
  8 x=-0.908 y= 0.687 (0.502)
  9 x=-0.908 y= 0.687 (0.502)
 10 x= 1.099 y=-0.688 (0.517)
Average fitness = 0.517

Iteration = 8
  1 x=-0.908 y= 0.687 (0.502)
  2 x=-0.908 y= 0.687 (0.502)
```

```
  3 x=-0.908 y= 0.687 (0.502)
  4 x=-0.908 y= 0.687 (0.502)
  5 x=-0.908 y= 0.687 (0.502)
  6 x=-0.908 y= 0.687 (0.502)
  7 x=-0.908 y= 0.687 (0.502)
  8 x=-0.908 y= 0.687 (0.502)
  9 x=-0.908 y= 0.687 (0.502)
 10 x= 1.099 y=-0.688 (0.517)
Average fitness = 0.504

Iteration = 9
  1 x=-0.908 y= 0.687 (0.502)
  2 x=-0.908 y= 0.687 (0.502)
  3 x=-0.908 y= 0.687 (0.502)
  4 x=-0.908 y= 0.687 (0.502)
  5 x=-0.908 y= 0.687 (0.502)
  6 x=-0.908 y= 0.687 (0.502)
  7 x=-0.908 y= 0.687 (0.502)
  8 x=-0.908 y= 0.687 (0.502)
  9 x=-0.908 y= 0.687 (0.502)
 10 x= 1.099 y=-0.688 (0.517)
Average fitness = 0.504

Iteration = 10
  1 x=-0.908 y= 0.687 (0.502)
  2 x=-0.908 y= 0.687 (0.502)
  3 x=-0.908 y= 0.687 (0.502)
  4 x=-0.908 y= 0.687 (0.502)
  5 x=-0.908 y= 0.687 (0.502)
  6 x=-0.908 y= 0.687 (0.502)
  7 x=-0.908 y= 0.687 (0.502)
  8 x=-0.908 y= 0.687 (0.502)
  9 x=-0.908 y= 0.687 (0.502)
 10 x= 1.099 y=-0.688 (0.517)
Average fitness = 0.504

Iteration = 11
  1 x=-0.908 y= 0.687 (0.502)
  2 x=-0.908 y= 0.687 (0.502)
  3 x=-0.908 y= 0.687 (0.502)
  4 x=-0.908 y= 0.687 (0.502)
  5 x=-0.908 y= 0.687 (0.502)
  6 x=-0.908 y= 0.687 (0.502)
  7 x=-0.908 y= 0.687 (0.502)
  8 x=-0.908 y= 0.687 (0.502)
  9 x=-0.908 y= 0.687 (0.502)
 10 x=-0.908 y= 0.687 (0.502)
Average fitness = 0.502
```

Вырождение популяции

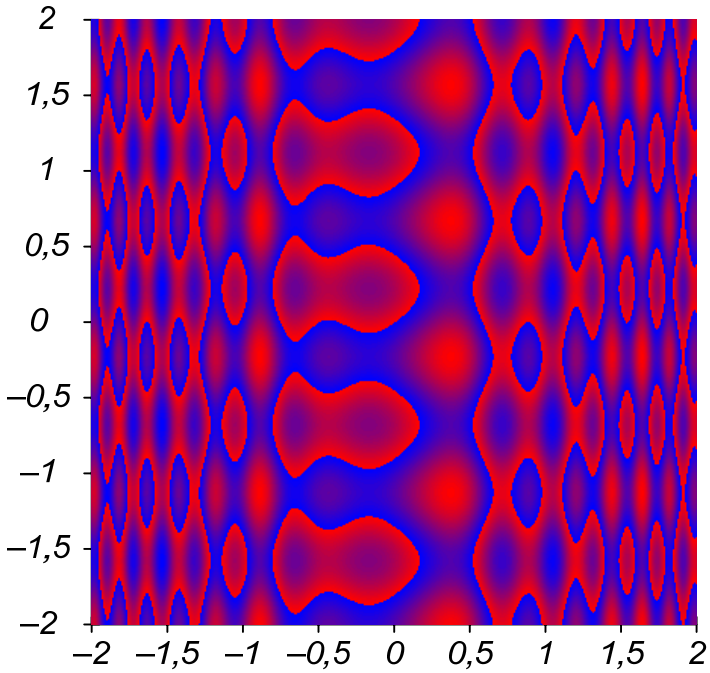
```
[skip]

Iteration = 37
  1 x=-0.908 y= 0.687 (0.502)
  2 x=-0.908 y= 0.687 (0.502)
  3 x=-0.908 y= 0.687 (0.502)
  4 x=-0.908 y= 0.687 (0.502)
  5 x=-0.908 y= 0.687 (0.502)
  6 x=-0.908 y= 0.687 (0.502)
  7 x=-0.908 y= 0.687 (0.502)
  8 x=-0.908 y=-0.416 (0.204)          ; !!!
  9 x=-0.908 y= 0.687 (0.502)
 10 x=-0.908 y= 0.687 (0.502)
Average fitness = 0.472

[skip]

Iteration = 73
  1 x=-0.908 y=-0.416 (0.204)
  2 x=-0.908 y=-0.416 (0.204)
  3 x=-0.908 y=-0.416 (0.204)
  4 x=-0.908 y=-0.416 (0.204)
  5 x=-0.914 y=-0.416 (0.201)          ; !!!
  6 x=-0.908 y=-0.416 (0.204)
  7 x=-0.908 y=-0.416 (0.204)
  8 x=-0.908 y=-0.416 (0.204)
  9 x=-0.908 y=-0.416 (0.204)
 10 x=-0.908 y=-0.416 (0.204)
Average fitness = 0.204

[skip]

Iteration = 96
  1 x=-0.908 y=-0.416 (0.204)          ; В популяции присутствуют разные особи
  2 x=-0.914 y=-0.416 (0.201)
  3 x=-0.914 y=-0.416 (0.201)
  4 x=-0.914 y=-0.416 (0.201)
  5 x=-0.914 y=-0.416 (0.201)
  6 x=-0.908 y=-0.416 (0.204)
  7 x=-0.914 y= 0.314 (0.126)          ; !!!
  8 x=-0.914 y=-0.416 (0.201)
  9 x=-0.914 y=-0.416 (0.201)
 10 x=-0.914 y=-0.416 (0.201)
Average fitness = 0.194

[skip]

Iteration = 106
  1 x=-0.914 y= 0.314 (0.126)          ; Популяция снова выродилась
```

```
  2 x=-0.914 y= 0.314 (0.126)
  3 x=-0.914 y= 0.314 (0.126)
  4 x=-0.914 y= 0.314 (0.126)
  5 x=-0.914 y= 0.314 (0.126)
  6 x=-0.914 y= 0.314 (0.126)
  7 x=-0.914 y= 0.314 (0.126)
  8 x=-0.914 y= 0.314 (0.126)
  9 x=-0.914 y= 0.314 (0.126)
 10 x=-0.914 y= 0.314 (0.126)
Average fitness = 0.126

[skip]

Iteration = 220
  1 x=-0.914 y= 0.314 (0.126)
  2 x=-0.914 y= 0.314 (0.126)
  3 x=-0.914 y= 0.314 (0.126)
  4 x=-0.914 y= 0.314 (0.126)
  5 x=-0.914 y= 0.314 (0.126)
  6 x=-0.914 y= 0.314 (0.126)
  7 x= 0.999 y= 0.314 (0.098)          ; !!!
  8 x=-0.914 y= 0.314 (0.126)
  9 x=-0.914 y= 0.314 (0.126)
 10 x=-0.914 y= 0.314 (0.126)
Average fitness = 0.123

[skip]

Iteration = 859
  1 x= 0.999 y= 0.314 (0.098)
  2 x= 0.999 y= 0.314 (0.098)
  3 x= 0.999 y= 0.314 (0.098)
  4 x= 0.999 y= 0.314 (0.098)
  5 x= 0.999 y= 0.314 (0.098)
  6 x= 0.999 y= 0.314 (0.098)
  7 x= 0.999 y= 0.314 (0.098)
  8 x= 0.999 y= 0.314 (0.098)
  9 x= 0.999 y= 0.107 (0.011)          ; !!!
 10 x= 0.999 y= 0.314 (0.098)
Average fitness = 0.090
[skip]

Iteration = 871
  1 x= 0.999 y= 0.107 (0.011)
  2 x= 0.999 y= 0.107 (0.011)
  3 x= 0.999 y= 0.107 (0.011)
  4 x= 0.999 y= 0.107 (0.011)
  5 x= 0.999 y= 0.107 (0.011)
  6 x= 0.999 y= 0.107 (0.011)
  7 x= 0.999 y= 0.107 (0.011)
  8 x= 0.999 y= 0.107 (0.011)
  9 x= 0.999 y= 0.041 (0.002)          ; !!!
 10 x= 0.999 y= 0.107 (0.011)
Average fitness = 0.010

[skip]

Iteration = 2701
  1 x= 0.999 y= 0.041 (0.002)
  2 x= 0.999 y= 0.041 (0.002)
  3 x= 0.999 y= 0.041 (0.002)
  4 x= 0.999 y= 0.041 (0.002)
  5 x= 0.999 y= 0.041 (0.002)
  6 x= 0.999 y= 0.041 (0.002)
  7 x= 0.999 y=-0.032 (0.001)          ; !!!
  8 x= 0.999 y= 0.041 (0.002)
  9 x= 0.999 y= 0.041 (0.002)
 10 x= 0.999 y= 0.041 (0.002)
Average fitness = 0.002

[skip]

Iteration = 4524
  1 x= 0.999 y=-0.032 (0.001)
  2 x= 0.999 y=-0.032 (0.001)
  3 x= 0.999 y=-0.032 (0.001)
  4 x= 0.999 y=-0.032 (0.001)
  5 x= 0.999 y=-0.032 (0.001)
  6 x= 0.999 y= 0.009 (0.000)          ; !!!
  7 x= 0.999 y=-0.032 (0.001)
  8 x= 0.999 y=-0.032 (0.001)
  9 x= 0.999 y=-0.032 (0.001)
 10 x= 0.999 y=-0.032 (0.001)
Average fitness = 0.001

[skip]

Iteration = 11597
  1 x= 0.999 y= 0.009 (0.000)
  2 x= 0.999 y= 0.009 (0.000)
  3 x= 0.999 y= 0.009 (0.000)
  4 x= 0.999 y= 0.009 (0.000)
  5 x= 0.999 y= 0.009 (0.000)
  6 x= 0.999 y= 0.009 (0.000)
  7 x= 0.999 y=-0.001 (0.000)          ; !!!
  8 x= 0.999 y= 0.009 (0.000)
  9 x= 0.999 y= 0.009 (0.000)
 10 x= 0.999 y= 0.009 (0.000)
Average fitness = 0.000
```

Пример 3 – функция с множеством экстремумов.

$f(x, y) = \sin 5x + \sin 7x + \sin 10x^2 \to \min,\ -2 \le x \le 2, -2 \le y \le 2.$

Множество локальных минимумов



Результаты эволюции (старт с разных начальных условий).

$r_0 = 0\ x = -1{,}537\ y = 1{,}564\ f = -2{,}982$

$r_0 = 1\ x = -1{,}537\ y = 1{,}564\ f = -2{,}982$

$r_0 = 2\ x = -1{,}539\ y = 0{,}676\ f = -2{,}980$

$r_0 = 3\ x = -1{,}537\ y = 0{,}671\ f = -2{,}983$

$r_0 = 4\ x = -1{,}538\ y = -0{,}229\ f = -2{,}981$

Эволюция приходит в разные локальные максимумы, близкие по значениям целевой функции