

Karolina Tatarczyk

Projekt 1 - Prosta Metoda Losowa (PML)

Funkcje testowe

Procedura

```

In[27]:= PML[function_, iterat_, number_, rangex_, rangey_] := Module[
  {dx = rangex, dy = rangey, f = function, m = iterat, n = number},
  maxVal = {};
  fAverVal = {};
  For[i = 1, i ≤ m, i++,
    tmpTable = {};
    FtmpTable = {};
    For[j = 1, j ≤ n, j++,
      AppendTo[tmpTable, {Random[Real, dx], Random[Real, dy]}]; (*losowanie n punktów*)
      AppendTo[FtmpTable, f[tmpTable[[j]]]];
      (*tablica z wartościami funkcji wylosowanych punktów*)
    ];
    bestpoint = tmpTable[[1]];
    For[k = 2, k ≤ n, k++,
      If[f[bestpoint] > f[tmpTable[[k]]], bestpoint = tmpTable[[k]];
      (*znalezienie punktu dla którego wartość f jest najmniejsza*)
    ];
    If[i == 1,
      AppendTo[maxVal, bestpoint],
      If[f[bestpoint] < f[maxVal[[i - 1]]],
        AppendTo[maxVal, bestpoint],
        AppendTo[maxVal, maxVal[[i - 1]]] (*warunek,
          że każdy kolejny punkt jest niegorszy od poprzedniego*)
      ];
    ];
    AppendTo[fAverVal, Mean[FtmpTable]]; (*średnia wartość w każdej iteracji*)
  ];
  fMaxVal = Table[f[maxVal[[i]]], {i, 1, m}];
  (*tablica z wartościami funkcji każdego najlepszego punktu*)
  Return[{maxVal, fMaxVal, fAverVal}]
]

```

Procedura wyświetlania

```

In[28]:= Drawing[function_, sol_, rangex_, rangey_] :=
  Module[{f = function, solution = sol, dx = rangex, dy = rangey},
    plot = Plot3D[f[{x, y}], {x, dx[[1]], dx[[2]]},
      {y, dy[[1]], dy[[2]]}, ColorFunction -> "BlueGreenYellow"];
    tabBestIteration = Table[{i, solution[[2, i]]}, {i, 1, it}];
    tabAverIteration = Table[{i, solution[[3, i]]}, {i, 1, it}];
    tabPoints = Table[Take[solution[[1]], i], {i, 1, it}];
    density = DensityPlot[f[{x, y}], {x, dx[[1]], dx[[2]]},
      {y, dy[[1]], dy[[2]]}, ColorFunction -> "BlueGreenYellow"];
    Linesplot = Table[ListLinePlot[tabPoints[[i]], PlotStyle -> Pink], {i, 1, it}];
    Pointsplot = Table[ListPlot[tabPoints[[i]], PlotStyle -> Pink], {i, 1, it}];
    animacja = ListAnimate[Table[Show[density, Linesplot[[i]], Pointsplot[[i]], {i, 1, it}],
      AnimationRunning -> False];

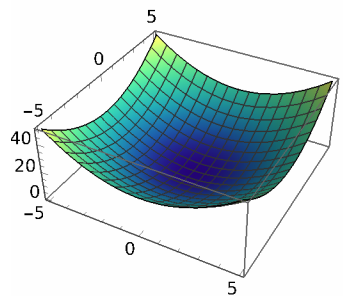
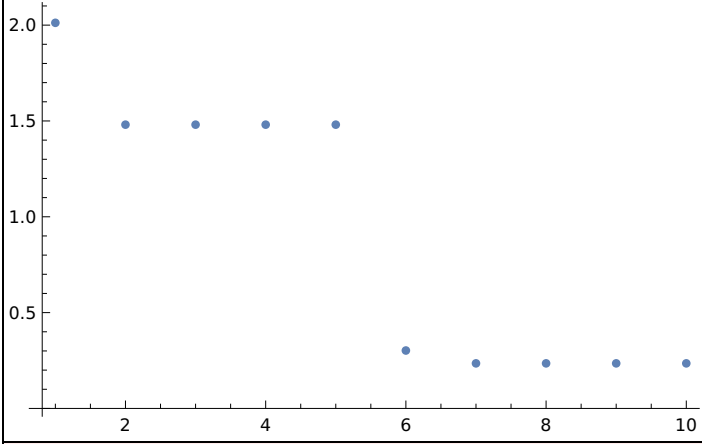
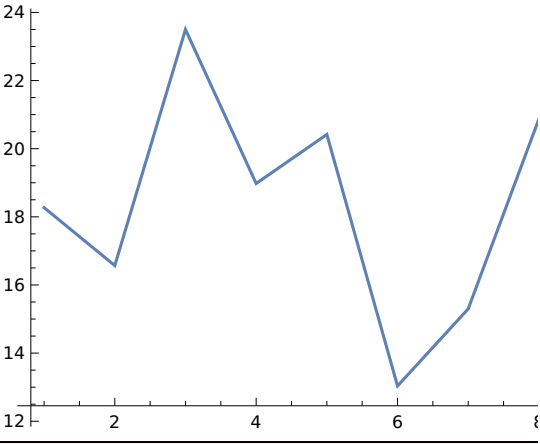
    iterBest = ListPlot[tabBestIteration, PlotRange -> Full, ImageSize -> 360];
    iterAver = ListLinePlot[tabAverIteration, PlotRange -> Full, ImageSize -> 360];
    Grid[{{"PROSTA METODA LOSOWA", SpanFromLeft},
      {"Wzór funkcji", "Wykres funkcji"}, {f[{x, y}], plot},
      {"Przebieg wartości minimalnej w zależności od iteracji",
        "Przebieg wartości średniej w zależności od iteracji"}, {iterBest, iterAver},
      {"Najlepsza znaleziona wartość", "Punkt, w którym znaleziono minimum"},
        {solution[[2, it]], solution[[1, it]]},
      {"Przebieg poszukiwania rozwiązania", SpanFromLeft}, {animacja, SpanFromLeft}},
    Frame -> All,
    Background -> {None, {Pink}, {{2, 1} -> LightPink, {2, 2} -> LightPink, {4, 1} -> LightPink,
      {4, 2} -> LightPink, {6, 1} -> LightPink, {6, 2} -> LightPink, {8, 1} -> LightPink}}]]

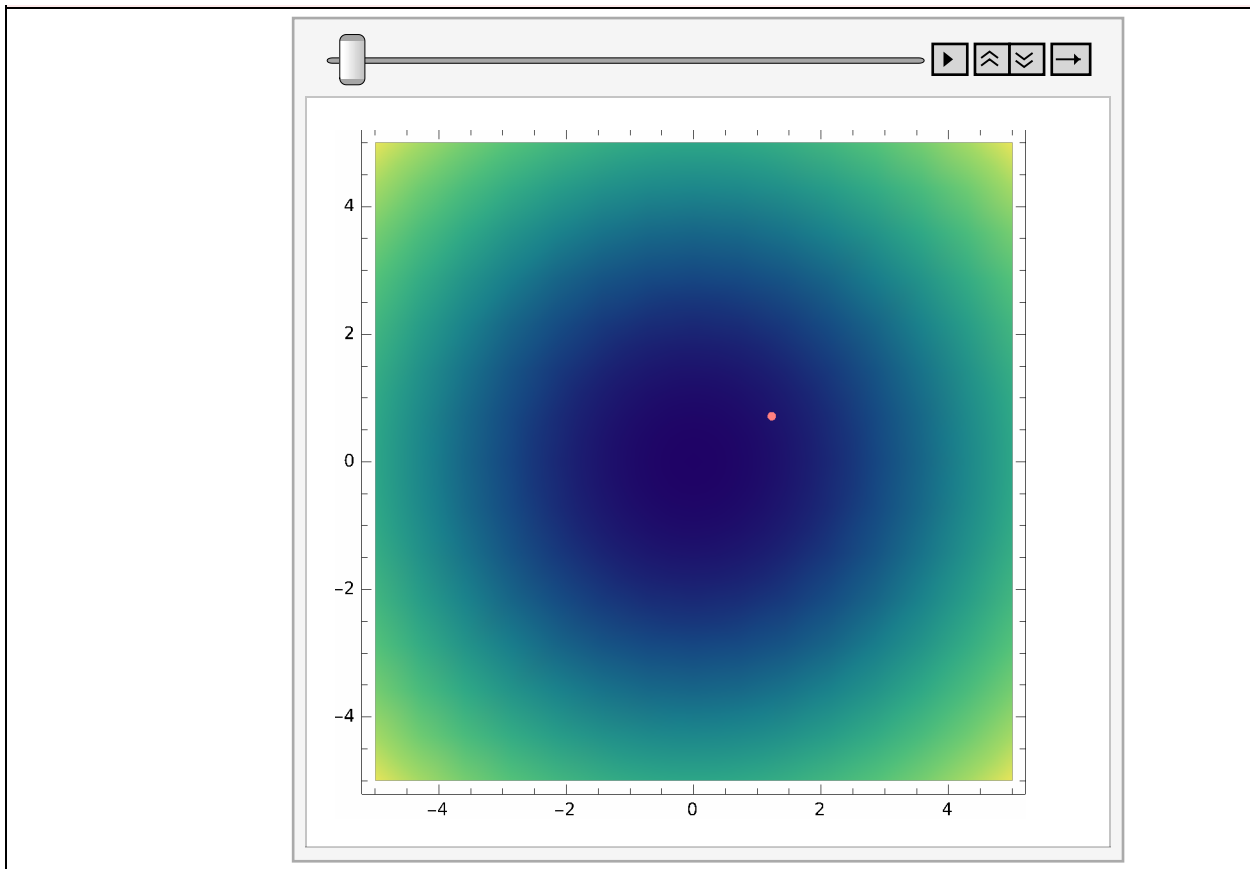
```

Przykład 1

```
In[29]:= Clear[f, dx, dy, n, it, solution]
f[{x_, y_}] := x^2 + y^2
dx = {-5, 5};
dy = {-5, 5};
n = 20;
it = 10;
solution = PML[f, it, n, dx, dy];
Drawing[f, solution, dx, dy]
```

Out[36]=

PROSTA METODA LOSOWA	
Wzór funkcji	Wykres funkcji
$x^2 + y^2$	
Przebieg wartości minimalnej w zależności od iteracji	Przebieg wartości średniej w zależności od ite
	
Najlepsza znaleziona wartość	Punkt, w którym znaleziono min
0.235219	{-0.435937, 0.21255}
Przebieg poszukiwania rozwiązania	



Przykład 2

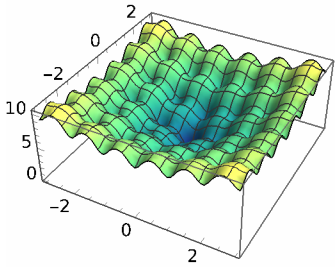
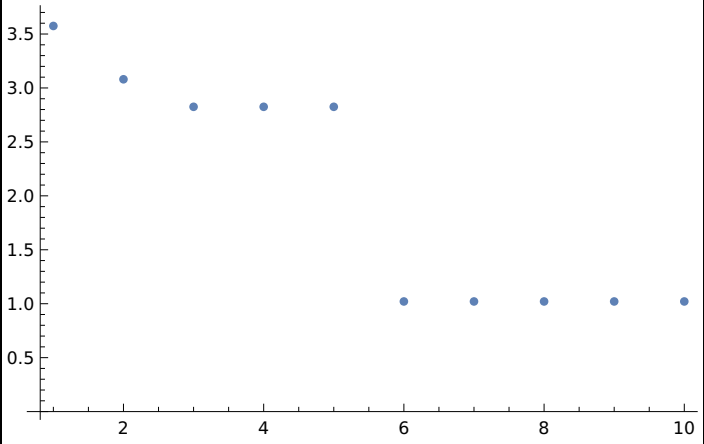
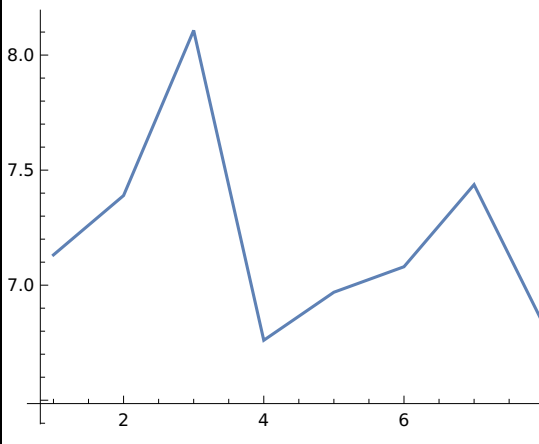
```

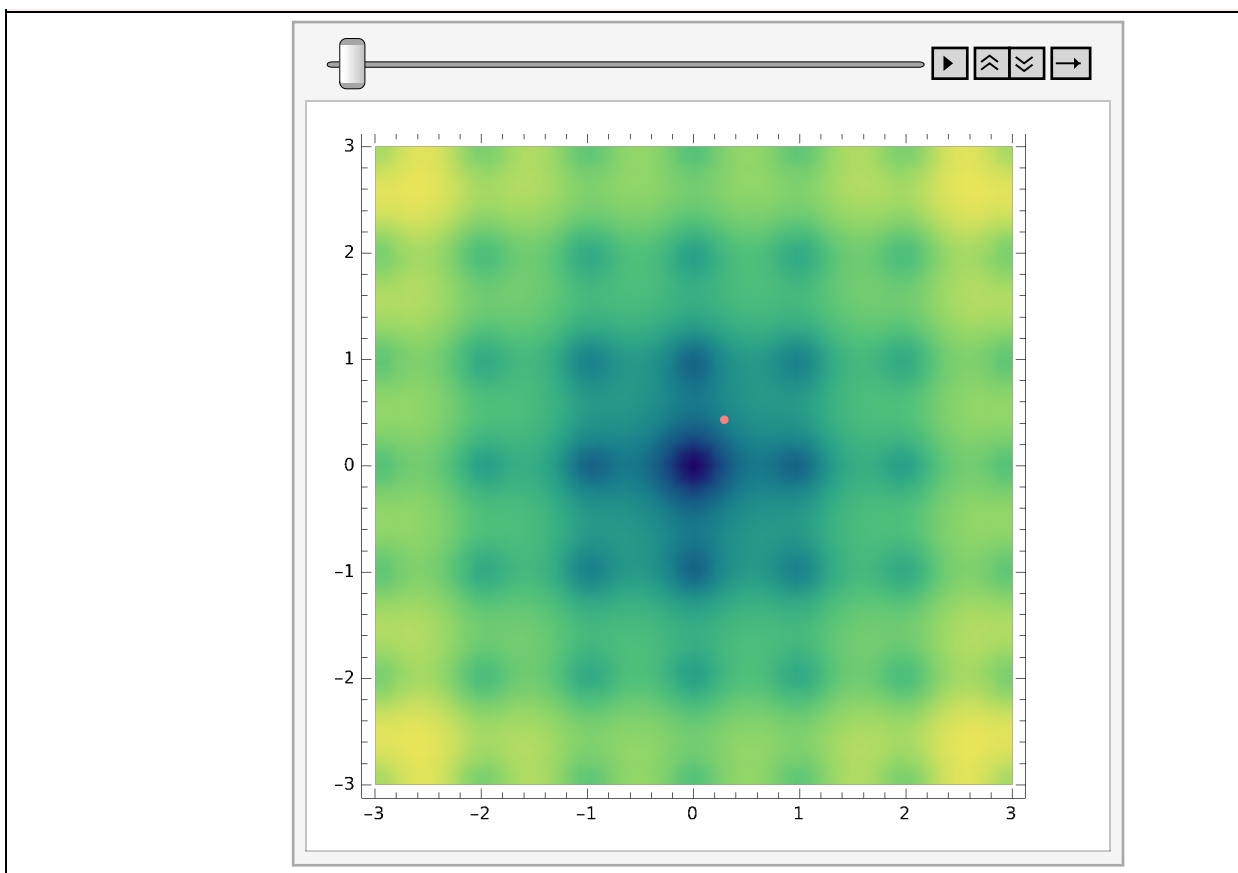
In[37]:= Clear[f, dx, dy, n, it, solution]
f[{x_, y_}] :=
  -20 * Exp[-0.2 * Sqrt[(x ^ 2 + y ^ 2) / 2]] - Exp[(Cos[2 * Pi * x] + Cos[2 * Pi * y]) / 2] + 20 + E
dx = {-3, 3};
dy = {-3, 3};
n = 20;
it = 10;
solution = PML[f, it, n, dx, dy];
Drawing[f, solution, dx, dy]

```

Out[44]=

PROSTA METODA LOSOWA	
Wzór funkcji	Wykres funkcji

$20 + e - 20 e^{-0.141421 \sqrt{x^2+y^2}} - e^{\frac{1}{2}} (\cos[2 \pi x] + \cos[2 \pi y])$																																													
Przebieg wartości minimalnej w zależności od iteracji	Przebieg wartości średniej w zależności od ite																																												
 <table border="1"> <caption>Data for Minimum Value Plot</caption> <thead> <tr> <th>Iteration</th> <th>Minimum Value</th> </tr> </thead> <tbody> <tr><td>1</td><td>3.5</td></tr> <tr><td>2</td><td>3.1</td></tr> <tr><td>3</td><td>2.8</td></tr> <tr><td>4</td><td>2.8</td></tr> <tr><td>5</td><td>2.8</td></tr> <tr><td>6</td><td>1.0</td></tr> <tr><td>7</td><td>1.0</td></tr> <tr><td>8</td><td>1.0</td></tr> <tr><td>9</td><td>1.0</td></tr> <tr><td>10</td><td>1.0</td></tr> </tbody> </table>	Iteration	Minimum Value	1	3.5	2	3.1	3	2.8	4	2.8	5	2.8	6	1.0	7	1.0	8	1.0	9	1.0	10	1.0	 <table border="1"> <caption>Data for Average Value Plot</caption> <thead> <tr> <th>Iteration</th> <th>Average Value</th> </tr> </thead> <tbody> <tr><td>1</td><td>7.1</td></tr> <tr><td>2</td><td>7.4</td></tr> <tr><td>3</td><td>8.1</td></tr> <tr><td>4</td><td>6.8</td></tr> <tr><td>5</td><td>7.0</td></tr> <tr><td>6</td><td>7.1</td></tr> <tr><td>7</td><td>7.4</td></tr> <tr><td>8</td><td>6.9</td></tr> <tr><td>9</td><td>6.8</td></tr> <tr><td>10</td><td>6.8</td></tr> </tbody> </table>	Iteration	Average Value	1	7.1	2	7.4	3	8.1	4	6.8	5	7.0	6	7.1	7	7.4	8	6.9	9	6.8	10	6.8
Iteration	Minimum Value																																												
1	3.5																																												
2	3.1																																												
3	2.8																																												
4	2.8																																												
5	2.8																																												
6	1.0																																												
7	1.0																																												
8	1.0																																												
9	1.0																																												
10	1.0																																												
Iteration	Average Value																																												
1	7.1																																												
2	7.4																																												
3	8.1																																												
4	6.8																																												
5	7.0																																												
6	7.1																																												
7	7.4																																												
8	6.9																																												
9	6.8																																												
10	6.8																																												
Najlepsza znaleziona wartość	Punkt, w którym znaleziono mi																																												
1.0208	{-0.0089332, -0.160936}																																												
Przebieg poszukiwania rozwiązania																																													

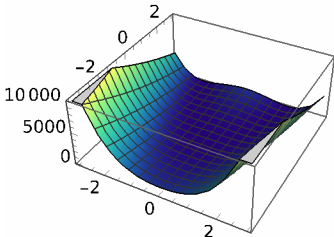
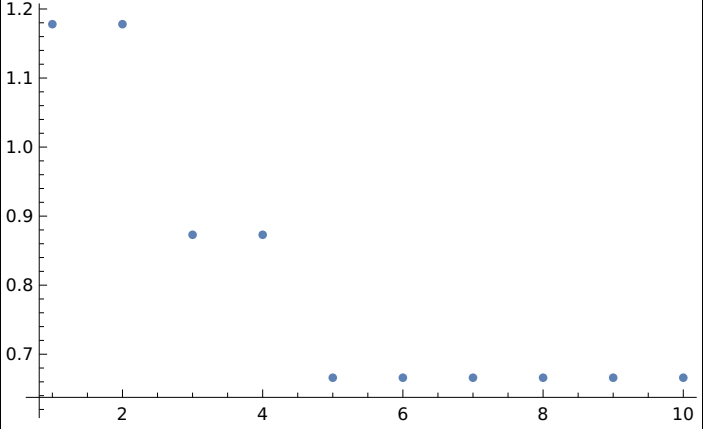
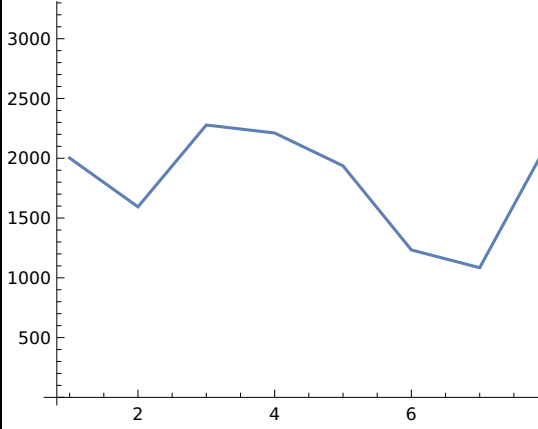


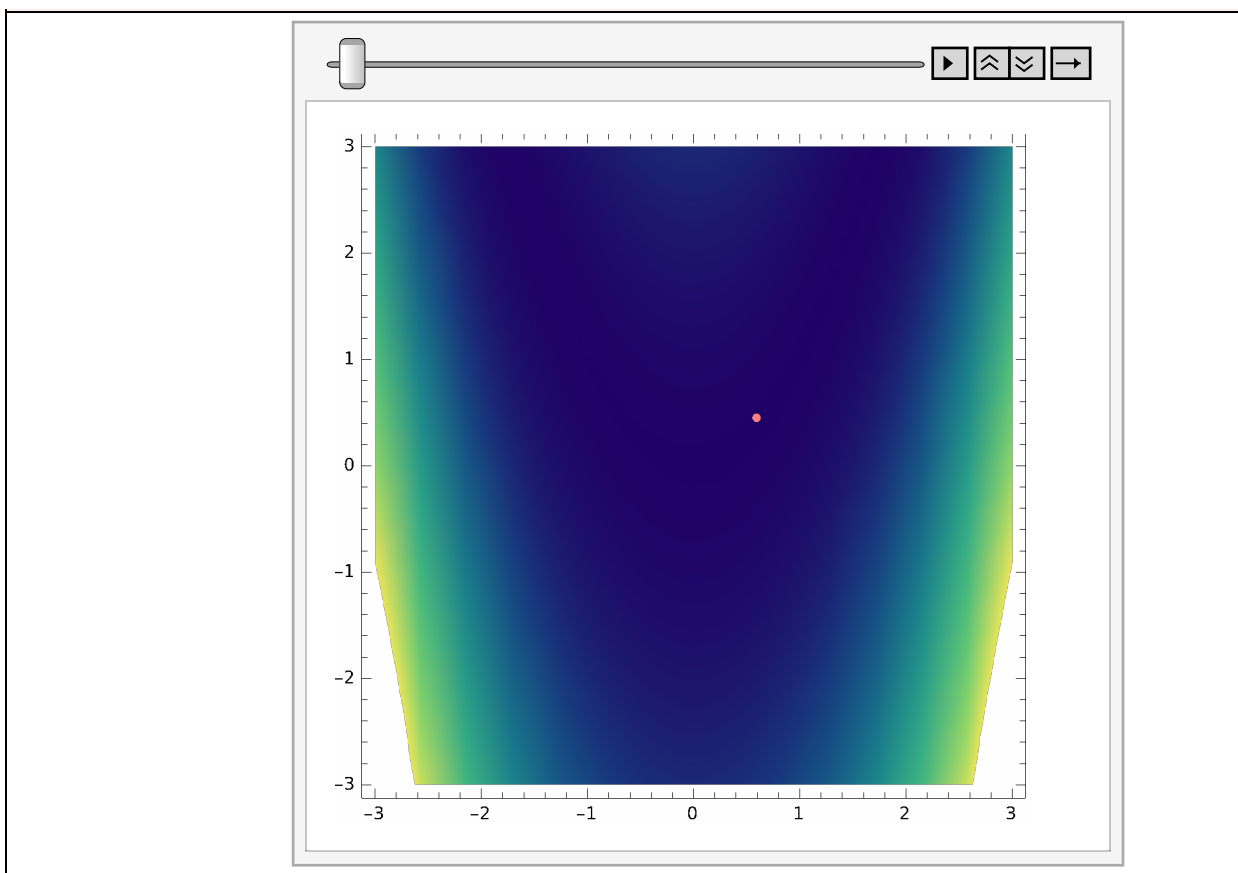
Przykład 3

```
In[45]:= Clear[f, dx, dy, n, it, solution]
f[{x_, y_}] := (1 - x)^2 + 100 (y - x^2)^2
dx = {-3, 3};
dy = {-3, 3};
n = 20;
it = 10;
solution = PML[f, it, n, dx, dy];
Drawing[f, solution, dx, dy]
```

Out[52]=

PROSTA METODA LOSOWA	
Wzór funkcji	Wykres funkcji

$(1 - x)^2 + 100(-x^2 + y)^2$	
Przebieg wartości minimalnej w zależności od iteracji	Przebieg wartości średniej w zależności od ite
	
Najlepsza znaleziona wartość	Punkt, w którym znaleziono min
0.66596	{0.299314, 0.131422}
Przebieg poszukiwania rozwiązania	



Wnioski

Prosta Metoda Losowa nie jest zbyt dobrą metodą na poszukiwanie minimum funkcji testowych.

Jak sama nazwa wskazuje metoda ta opiera się na losowaniu. Prostota algorytmu oraz fakt, iż punkty są losowane z całego przedziału nie zapewnia nam uzyskania faktycznej wartości minimalnej. Jedynie warunek iż punkty w każdej kolejnej iteracji muszą być niegorsze zapewnia nam pewną stabilność wyników. Zwiększenie punktów losowanych w każdej iteracji może zwiększyć szanse na znalezienie wartości coraz to lepszych.

Średnia uzyskiwana wartość jest losowa, więc nie jesteśmy w stanie znaleźć żadnych zależności związanych z kolejnymi iterac-

jami.

Zwiększenie liczby iteracji bardzo wydłuża czas obliczeń wykonanych przez program.