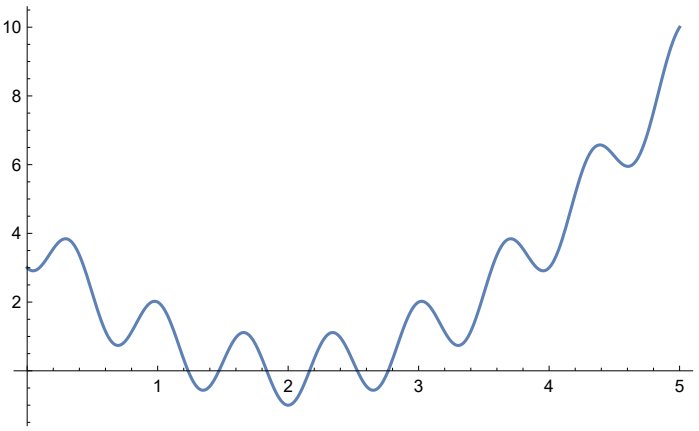
Finding a minimum of a function

Mathematica solution

In[204]:=

```
f[x_] := (x - 2)^2 - Cos[3 π (x - 2)];
Plot[f[x], {x, 0, 5}]
FindMinimum[f[x], x]
FindMinimum[f[x], {x, 1}]
FindMinimum[f[x], {x, 3}]
FindMinimum[f[x], {x, 4}]
FindMinimum[f[x], {x, 3.9}]
FindMinimum[f[x], {x, 2.2}]
```

Out[205]=



Out[206]=

```
{-0.565358, {x → 1.34806}}
```

Out[207]=

```
{-0.565358, {x → 1.34806}}
```

Out[208]=

```
{-0.565358, {x → 1.34806}}
```

Out[209]=

```
{-1., {x → 2.}}
```

Out[210]=

```
{2.91062, {x → 3.95462}}
```

Out[211]=

```
{-1., {x → 2.}}
```

Approximative solution using simulated annealing

In[212]:=

```
SimulatedAnnealingToFindMinimum[f_, x0_, d_, temperatures_, nIter_] :=
  (*
  input:
    f = function to minimize,
    x0 = initial guess where minimum lies,
    d = characteristic step to make in each iteration,
    temperatures = array of temperatures to use in decreasing order,
    nIter = number of iterations to do for each temperature,
  output:
    foundMinima = a list of found minima for each temperature
  *)
  Module[
   {nT, foundMinima, x, fx, it, T, i, xtrial, ftrial, xLocalMin, fLocalMin, rejected},
   nT = Length[temperatures];
   foundMinima = ConstantArray[0, nT];
   x = x0; (* initial guess *)
   fx = f[x]; (* initial value *)
   (* temperature loop *)
   Do[
    T = temperatures[[it]]; (* current temperature *)
    (* Monte Carlo iterations *)
    rejected = 0;
    xLocalMin = x;
    fLocalMin = fx;
    Do[
     (* make a random step *)
     xtrial = x + RandomReal[{-d, d}];
     ftrial = f[xtrial];
     If[ftrial < fx || RandomReal[] < Exp[(fx - ftrial) / T],
      x = xtrial;
      fx = ftrial,
      rejected++
     ];
     If[fx < fLocalMin,
      xLocalMin = x;
      fLocalMin = fx
     ],
     {i, 1, nIter}
    ];
    foundMinima[[it]] = {xLocalMin, fLocalMin},
    {it, 1, nT}
   ];
   Return[foundMinima]
  ];
```

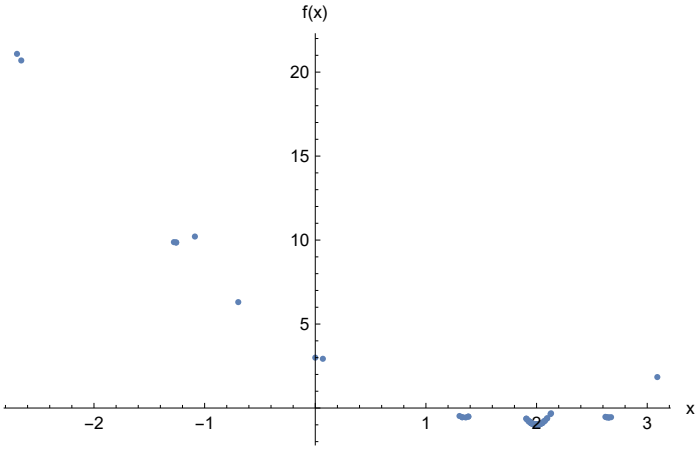
In[317]:=

```
Ts = Sort[Union[Range[1.0, 10.0, 1.0], Range[0.01, 1.0, 0.01]], Greater];
x0 = 0.0;
δ = 1.0;
nIter = 10;
points = SimulatedAnnealingToFindMinimum[f, x0, δ, Ts, nIter];
Print["Starting point: x = ", x0, ", f(x) = ", f[x0]];
Print["Last minimum: x = ", Last[points][[1]], ", f(x) = ", Last[points][[2]];
ListPlot[points, AxesLabel → {"x", "f(x)"}, PlotRange → All]
ListPlot[Transpose[points], AxesLabel → {"n", "x,f(x)"}, PlotLegends → {"x", "fx"}]
```

```
Starting point: x = 0., f(x) = 3.

Last minimum: x = 2.01014, f(x) = -0.995332
```

Out[324]=



Out[325]=