

```

GeneracjaPopulacji[wielkoscPopulacji_, iloscWymiarow_, ograniczenia_] :=
Module[{λ = wielkoscPopulacji, wym = iloscWymiarow, populacja},
moduł
    populacja = Table[Table[RandomReal[{ograniczenia[[j, 1]], ograniczenia[[j, 2]]}],
tabela tabela losowa liczba rzeczywista
        {j, 1, wym}], {i, 1, λ}];
    Return[populacja]
zwróć

```

```

Reprodukcja[q_, populacjaBazowa_,
    funkcjaOceny_, wielkoscPopulacjiBazowej_, iloscWymiarow_] :=
Module[{zmienne, λ = wielkoscPopulacjiBazowej, populacjaTymczasowa},
moduł
    turniej, indeksNajlepszegoOsobnika, wartosciTurnieju},
    Clear[zmienne, wartosciTurnieju, turniej, indeksNajlepszegoOsobnika];
wyczyść
    populacjaTymczasowa = {};
    zmienne = Table[Symbol[StringJoin["x", ToString[i]]], {i, 1, iloscWymiarow}];
tabela symbol połącz ciągi znaków przemień na ciąg znaków
    Do[turniej = Table[populacjaBazowa[RandomInteger[{1, λ}]]], {i, 1, q}];
rób tabela losowa liczba całkowita
    wartosciTurnieju = Table[funkcjaOceny, {i, 1, q}];
tabela
    Do[Do[wartosciTurnieju[[k]] =
rób rób
        wartosciTurnieju[[k]] /. {zmienne[[i]] → turniej[[k, i]]}
        , {i, 1, iloscWymiarow}];
        , {k, 1, q}];
    indeksNajlepszegoOsobnika = Position[wartosciTurnieju, Min[wartosciTurnieju]];
pozycja minimum
    AppendTo[populacjaTymczasowa, turniej[[indeksNajlepszegoOsobnika[[1, 1]]]]];
dołącz na końcu do wartości zmiennej
    , {j, 1, λ}];
    Return[populacjaTymczasowa]
zwróć

```

```

Krzyzowanie[osobnikRodzicielski1_, osobnikRodzicielski2_, iloscWymiarow_] :=
Module[{wym = iloscWymiarow, osobnikPotomny1, osobnikPotomny2, punktPrzeciecia},
moduł
    punktPrzeciecia = RandomInteger[{1, wym}];
losowa liczba całkowita
    osobnikPotomny1 = Flatten[{Part[osobnikRodzicielski1, 1 ;; punktPrzeciecia],
splaszcz część
        Part[osobnikRodzicielski2, punktPrzeciecia + 1 ;;]}];
część
    osobnikPotomny2 = Flatten[{Part[osobnikRodzicielski2, 1 ;; punktPrzeciecia],
splaszcz część
        Part[osobnikRodzicielski1, punktPrzeciecia + 1 ;;]}];
część
    Return[{osobnikPotomny1, osobnikPotomny2}]
zwróć

```

```

MutacjaZParametrem[mutowanyOsobnik_, parametr_, iloscWymiarow_] :=
Module[{ $\sigma$  = parametr,  $\xi$ , x = mutowanyOsobnik},
  |moduł
   $\xi$  = RandomVariate[NormalDistribution[0,  $\sigma$ ], iloscWymiarow];
  |próba losowa |rozkład normalny
  x +=  $\xi$ ;
  Return[x];
  |zwróć

MutacjaDeterministyczna[mutowanyOsobnik_,
  iloscWymiarow_, numerGeneracji_, maksymalnaIloscGeneracji_] :=
Module[{ $\xi$ ,  $\sigma$ , t = numerGeneracji, T = maksymalnaIloscGeneracji, x = mutowanyOsobnik},
  |moduł
   $\sigma$  = 1 - 0.9 * t / T;
   $\xi$  = RandomVariate[NormalDistribution[0,  $\sigma$ ], iloscWymiarow];
  |próba losowa |rozkład normalny
  x +=  $\xi$ ;
  Return[x];
  |zwróć

MutacjaAdaptacyjna[mutowanyOsobnik_,
  iloscWymiarow_, numerGeneracji_, czestoscSukcesow_] :=
Module[{ $\xi$ ,  $\sigma$  = 0.5, x = mutowanyOsobnik, n = 10, p = czestoscSukcesow, c = 0.82},
  |moduł
  If[numerGeneracji % 10 == 0, If[p > 1 / 5,  $\sigma$  =  $\sigma$  / c,  $\sigma$  =  $\sigma$  * c]];
  |operator warunkowy |operator warunkowy
   $\xi$  = RandomVariate[NormalDistribution[0,  $\sigma$ ], iloscWymiarow];
  |próba losowa |rozkład normalny
  x +=  $\xi$ ;
  Return[x];
  |zwróć

MutacjaSamoadaptująca[mutowanyOsobnik_, iloscWymiarow_] :=
Module[{ $\tau$  = 1 / Sqrt[iloscWymiarow], x = mutowanyOsobnik,  $\sigma$ ,  $\xi$ },
  |moduł |pierwiastek kwadratowy
   $\sigma$  = x[iloscWymiarow];
   $\sigma$  *= E^RandomVariate[NormalDistribution[0,  $\tau$ ]];
  |I· |próba losowa |rozkład normalny
  x[iloscWymiarow] =  $\sigma$ ;
   $\xi$  = RandomVariate[NormalDistribution[0,  $\sigma$ ], iloscWymiarow - 1];
  |próba losowa |rozkład normalny
  AppendTo[{ $\xi$ , 0}];
  |dołącz na końcu do wartości zmiennej
  x +=  $\xi$ ;
  Return[x];
  |zwróć

```

```

ZnalezienieNajlepszegoOsobnika[populacja_,
  funkcjaCelu_, iloscWymiarow_, wielkoscPopulacji_] :=
Module[{zmienne, wartosciFunkcji, λ = wielkoscPopulacji, indeksNajlepszegoOsobnika},
  (*moduł*)
  zmienne = Table[Symbol[StringJoin["x", ToString[i]]], {i, 1, iloscWymiarow}];
  (*tabela symbol połącz ciągi znaków przemień na ciąg znaków*)
  wartosciFunkcji = Table[funkcjaCelu, {i, 1, λ}];
  (*tabela*)

  Do[Do[
    (*rób*)
    wartosciFunkcji[[k]] = wartosciFunkcji[[k]] /. {zmienne[[i]] → populacja[[k, i]]}
    , {i, 1, iloscWymiarow}];
    , {k, 1, λ}];
  indeksNajlepszegoOsobnika =
    Position[wartosciFunkcji, Min[wartosciFunkcji]][[1, 1]];
  (*pozycja minimum*)
  Return[{populacja[indeksNajlepszegoOsobnika]},
    (*zwróć*)
    wartosciFunkcji[indeksNajlepszegoOsobnika]]];
]

AGzParametrem[licznoscPopulacji_, funkcjaCelu_,
  q_, iloscWymiarow_, ograniczenia_, prawdopodobienstwoMutacji_,
  prawdopodobienstwoKrzyzowania_, przewidywaneOptimum_, parametr_] :=
Module[{λ = licznoscPopulacji, f = funkcjaCelu, populacjaBazowa, populacjaPotomna},
  (*moduł*)
  pm = prawdopodobienstwoMutacji, pk = prawdopodobienstwoKrzyzowania,
  populacjaTymczasowa, najlepszyOsobnik, numerGeneracji},
  numerGeneracji = 1;
  populacjaBazowa = GeneracjaPopulacji[λ, iloscWymiarow, ograniczenia];
  najlepszyOsobnik =
    ZnalezienieNajlepszegoOsobnika[populacjaBazowa, f, iloscWymiarow, λ];
  While[Abs[najlepszyOsobnik[[2]] - przewidywaneOptimum] > 10-4,
    (*podciąg wartość bezwzględna*)
    populacjaPotomna = Reprodukacja[q, populacjaBazowa, f, λ, iloscWymiarow];
    (*Krzyżowanie z prawdopodobienstwem*)
    Do[If[RandomReal[] ≤ pm, populacjaPotomna[[i]] =
      (*rób*) (*losowa liczba rzeczywista*)
      MutacjaZParametrem[populacjaPotomna[[i]], parametr, iloscWymiarow]]
      , {i, 1, λ}];
    populacjaBazowa = populacjaPotomna;
    najlepszyOsobnik =
      ZnalezienieNajlepszegoOsobnika[populacjaBazowa, f, iloscWymiarow, λ];
    numerGeneracji++;
  ];
  Print[numerGeneracji];
  (*drukuj*)
  Return[najlepszyOsobnik]
  (*zwróć*)
]

(*TESTY*)

pop = GeneracjaPopulacji[500, 2, {{-5, 5}, {-5, 5}}];

```

```
ZnalezienieNajlepszegoOsobnika[pop, x1^2 + x2^2, 2, 500]
```

```
{ {-0.38597, 0.0608789}, 0.152679 }
```

```
Reprodukcja[50, pop, x1^2 + x2^2, 500, 2];
```

```
AGzParametrem[500, x1^2 + x2^2, 10, 2, {{-10, 10}, {-10, 10}}, 0.4, 0, 0, 1]
```

```
92
```

```
{ {0.00339844, -0.00754992}, 0.0000685507 }
```