ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

**БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ**

**ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ**

(НИУ «БелГУ»)

ИНСТИТУТ ИНЖЕНЕРНЫХ И ЦИФРОВЫХ ТЕХНОЛОГИЙ

КАФЕДРА МАТЕМАТИЧЕСКОГО И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ

**Отчет по лабораторной работе 4**

**по дисциплине: «Интеллектуальные системы»**

студента очного отделения

4 курса 12001801 группы

Капустина Виктора Сергеевича

Проверил(а):

Петрова Елена Вадимовна

Белгород 2022

## **Лабораторная работа №4**

**Цель работы:**

Приобретение и закрепление знаний, и получение практических навыков работы с генетическими алгоритмами.

**Постановка задачи:**

1. Разработайте генетический алгоритм для решения уравнения (4 параметра).

2. Для заданного варианта расписать пошаговое использование генетического алгоритма

3. Разработайте программу, реализующую генетический алгоритм.

4. Оформить отчет по лабораторной работе.

| Вариант | Уравнение |
| --- | --- |
| 4 | 2a-2b-c+4d=27 |

**Выполнение работы:**

Дано диофантово уравнение 2a-2b-c+4d=27. Необходимо найти целые решения этого уравнения.

Для того для значений a, b, c и d подставляются случайные значения с учетом ограничений (1 ≤ a, b, c, d ≤ 27).

**Первое поколение**

Выберем начальное поколение хромосом путем генерации каждой переменной из пяти решений:

Таблица 1 – Начальное поколение хромосом

| Хромосома | (a, b, c, d) |
| --- | --- |
| 1 | (22, 3, 4, 18) |
| 2 | (13, 10, 7, 15) |
| 3 | (6, 18, 13, 16) |
| 4 | (9, 13, 4, 19) |
| 5 | (13, 17, 17, 3) |

После чего высчитываем функцию приспособленности, подставляя каждое решение в выражение 2a-2b-c+4d=27. Расстояние от полученного значения до 27 и будет считать целевым значением:

Таблица 2 – Функция приспособленности для начального поколения хромосом

| Хромосома | Функция приспособленности |
| --- | --- |
| 1 | |106 - 27| = 79 |
| 2 | |59 - 27| = 32 |
| 3 | |27- 27| = 0 |
| 4 | |64-27| = 37 |
| 5 | |(-13)-27| = -40 |

Если бы на этом шаге результат бы не был найден. Следующим шагом мы бы высчитывали вероятность хромосомы быть родителем. Для этого возьмем сумму обратных значений функций.

Таблица 3 – Вероятность оказаться родительской хромосомой

| Хромосома | Вероятность оказаться родительской хромосомой |
| --- | --- |
| 1 | (1/p1)/⅀p\*100=x1% |
| 2 | (1/p2)/ ⅀p\*100=x2% |
| 3 | (1/p3)/ ⅀p\*100=x3% |
| 4 | (1/p4)/⅀p\*100=x4% |
| 5 | (1/p5)/⅀p\*100=x5% |

Для одной пары необходимо выбрать 2 различные хромосомы. Таким образом можем представить следующую таблицу выбора родителей. Возьмем случайные значения и подставим для моделирования развития дальнейшего расчета.

Таблица 4 – Моделирование выбора родителей

| Хромосома-отец | Хромосома-мать |
| --- | --- |
| 3 | 1 |
| 4 | 1 |
| 3 | 5 |
| 4 | 2 |
| 1 | 4 |

**Второе поколение**

Таблица 5 – Скрещивание хромосом родителей

| Хромосома-отец | Хромосома-мать | Хромосома-потомок |
| --- | --- | --- |
| (6 | 18, 13, 16) | (22 | 3, 4, 18) | (6, 3, 4, 18) |
| (9, 13 | 4, 19) | (22, 3 | 4, 18) | (9, 13, 4, 18) |
| (6, 18, 13 | 16) | (13, 17, 17 | 3) | (6, 18, 13, 3) |
| (9 | 13, 4, 19) | (13 | 10, 7, 15) | (9, 10, 7, 15) |
| (22, 3 | 4, 18) | (9, 13 | 4, 19) | (22, 3, 4, 19) |

Далее пришлось бы итерационно считать функции приспособленности для заданного выражения, пока бы функция приспособления не оказалась равна 0. Т.к. результат достигнут был достигнут при первом поколении, тут и остановимся.

  
Рис. 1 Результат работы программы

**Приложение**

**Листинг программы 1**

#include <iostream>

#include <ctime>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Rus");

srand(time(0));

int n=4;

int \*arr1 = new int[n];

int \*arr2 = new int[n];

int \*arr3 = new int[n];

int \*arr4 = new int[n];

int \*arr5 = new int[n];

for(int i=0;i<n;i++)

{

arr1[i] = 1 + rand() % 27;

arr2[i] = 1 + rand() % 27;

arr3[i] = 1 + rand() % 27;

arr4[i] = 1 + rand() % 27;

arr5[i] = 1 + rand() % 27;

}

cout << " Хромосома 1 ";

for (int i = 0; i < n; i++)

{

cout << arr1[i] << " ";

}cout << endl;

cout << " Хромосома 2 ";

for (int i = 0; i < n; i++)

{

cout<< arr2[i] << " ";

}cout << endl;

cout << " Хромосома 3 ";

for (int i = 0; i < n; i++)

{

cout << arr3[i] << " ";

}cout << endl;

cout << " Хромосома 4 ";

for (int i = 0; i < n; i++)

{

cout << arr4[i] << " ";

}cout << endl;

cout << " Хромосома 5 ";

for (int i = 0; i < n; i++)

{

cout << arr5[i] << " ";

}cout << endl;

double\* tempF = new double[5];

int\* Father1 = new int[5];

int\* Mother1 = new int[5];

int\* Father2 = new int[5];

int\* Mother2 = new int[5];

int\* Father3 = new int[5];

int\* Mother3 = new int[5];

int\* Father4 = new int[5];

int\* Mother4 = new int[5];

int\* Father5 = new int[5];

int\* Mother5 = new int[5];

int flag = 1;

int count = 0;

while ((flag != 0)&(count!=7))

{

tempF[0] = (2 \* arr1[0] - 2 \* arr1[1] - 1 \* arr1[2] + 4 \* arr1[3])-27;

tempF[1] = (2 \* arr2[0] - 2 \* arr2[1] - 1 \* arr2[2] + 4 \* arr2[3])-27;

tempF[2] = (2 \* arr3[0] - 2 \* arr3[1] - 1 \* arr3[2] + 4 \* arr3[3])-27;

tempF[3] = (2 \* arr4[0] - 2 \* arr4[1] - 1 \* arr4[2] + 4 \* arr4[3])-27;

tempF[4] = (2 \* arr5[0] - 2 \* arr5[1] - 1 \* arr5[2] + 4 \* arr5[3])-27;

for (int i = 0; i < n; i++)

{

if (tempF[i] == 0) {

flag = 0;

cout << endl << " Результат найден" << endl;

}

}

double tempInt = 0;

for (int i = 0; i < 5; i++)

{

tempInt += 1 / tempF[i];

}

for (int j = 0; j < n; j++)

{

Father1[j] = arr3[j];

Mother1[j] = arr1[j];

Father2[j] = arr4[j];

Mother2[j] = arr1[j];

Father3[j] = arr3[j];

Mother3[j] = arr5[j];

Father4[j] = arr4[j];

Mother4[j] = arr2[j];

Father5[j] = arr1[j];

Mother5[j] = arr4[j];

}

int i = 0;

arr1[i] = Father1[i];

arr2[i] = Father2[i];

arr3[i] = Father3[i];

arr4[i] = Father4[i];

arr5[i] = Father5[i];

i = 1;

arr1[i] = Mother1[i];

arr2[i] = Father2[i];

arr3[i] = Father3[i];

arr4[i] = Mother4[i];

arr5[i] = Father5[i];

i = 2;

arr1[i] = Mother1[i];

arr2[i] = Mother2[i];

arr3[i] = Father3[i];

arr4[i] = Mother4[i];

arr5[i] = Mother5[i];

i = 3;

arr1[i] = Mother1[i];

arr2[i] = Mother2[i];

arr3[i] = Mother3[i];

arr4[i] = Mother4[i];

arr5[i] = Mother5[i];

cout << " Функция приспособленности " << endl;

for (int i = 0; i < 5; i++)

{

cout <<tempF[i] << " ";

}

cout << endl;

//cout << tempInt << " Int " << endl;

cout << endl << " Хромосома 1 " << endl;

for (int i = 0; i < n; i++)

{

cout << arr1[i] << " ";

}

cout << endl << " Хромосома 2 " << endl;

for (int i = 0; i < n; i++)

{

cout << arr2[i] << " ";

}

cout << endl << " Хромосома 3 " << endl;

for (int i = 0; i < n; i++)

{

cout << arr3[i] << " ";

}

cout << endl << " Хромосома 4 " << endl;

for (int i = 0; i < n; i++)

{

cout << arr4[i] << " ";

}

cout << endl << " Хромосома 5 "<< endl;

for (int i = 0; i < n; i++)

{

cout << arr5[i] << " ";

}

count++;

cout << "Поколение - " << count << endl;

}

return 0;

}

**Конец листинга 1**