# Artyom Yudenkov

# TASK – 5

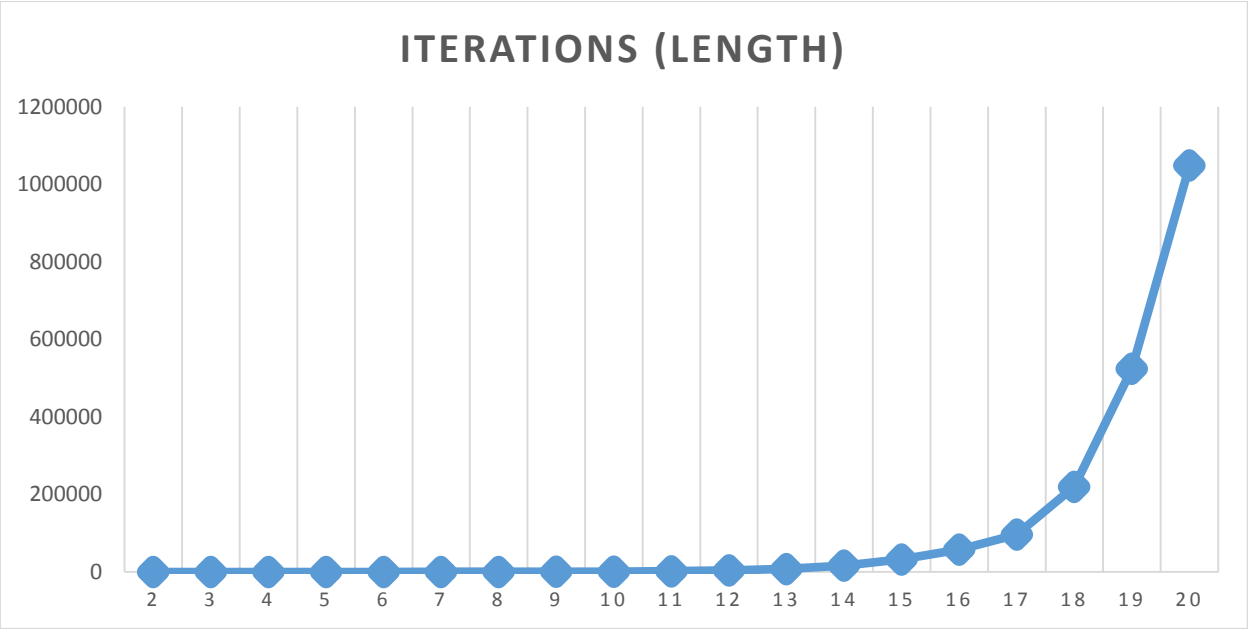
**The Task is :**

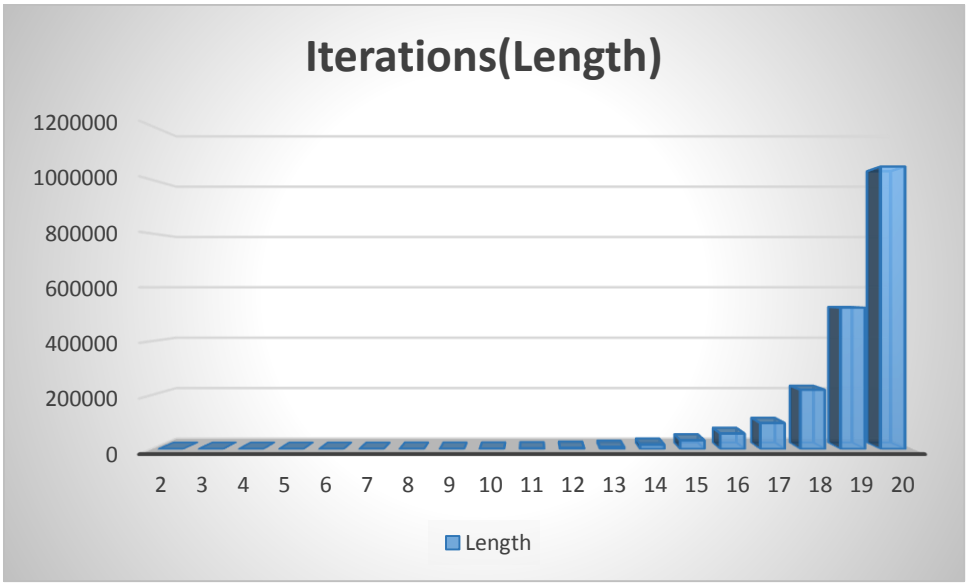
**1: Generate 2-20 bits pin code;**

**2: Find this pin using random;**

**3: Calculate the number of steps to find success result for each pin;**

| Length | Iterations (average tries) |
|---|---|
| 2 | 2 |
| 3 | 6 |
| 4 | 14 |
| 5 | 28 |
| 6 | 52 |
| 7 | 121 |
| 8 | 213 |
| 9 | 479 |
| 10 | 874 |
| 11 | 1986 |
| 12 | 3993 |
| 13 | 7638 |
| 14 | 16296 |
| 15 | 32547 |
| 16 | 57522 |
| 17 | 96387 |
| 18 | 220078 |
| 19 | 524201 |
| 20 | 1048485 |




**The code that helped us to do it :**

**Class.cs**

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
```

```
namespace AI{
  public class Class  {
    List<int> chromosoma = new List<int>();
    List<int> steps = new List<int>();
    Random rnd = new Random();
    public Class()    {
      chromosoma.Add(rnd.Next(0, 2));
      chromosoma.Add(rnd.Next(0, 2));
      var item = new List<int>();
      for (int j = 0; j < 10; j++)      {
        int step = 1;
        while (true)        {
          for (int i = 0; i < chromosoma.Count; i++)          {
            item.Add(rnd.Next(0, 2));          }
          if (item.SequenceEqual(chromosoma))          {
            break;
          }
          else
          {
            item = new List<int>();
            step++;         }        }
        steps.Add(step);      }    }
    public int GeAverage()    {
      return (int)steps.Average();    }
    public void Generation()    {
      var item = new List<int>();
      chromosoma.Add(rnd.Next(0, 2));
      for (int j = 0; j < 10; j++)      {
        int step = 1;
        while (true)
        {
```

```
        for (int i = 0; i < chromosoma.Count; i++)          {
          item.Add(rnd.Next(0, 2));          }
        if (item.SequenceEqual(chromosoma))          {
          break;
        }
        else
        {
          item = new List<int>();
          step++;          }        }
      steps.Add(step);     }    }
  }
}
```

## Program.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace AI{
  class Program  {
    static void Main(string[] args)    {
      List<int> average = new List<int>();
      Class cl = new Class();
      int i = 0;
      average.Add(cl.GeAverage());
      while(i++ < 19)
      {
        cl.Generation();
        average.Add(cl.GeAverage());
```

```
        Console.WriteLine(i);
      }
      for (int j = 0; j < average.Count; j++)
      {
        Console.WriteLine(average[j]);
      }
    }
  }
}
```