МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А.И. ГЕРЦЕНА»



Направление подготовки

09.03.01 – Информатика и вычислительная техника

Профиль «Технологии разработки программного обеспечения»

**Индивидуальное задание**

**“Аппроксимация числа Пи методом Бюффона”**

|  | Работу выполнили:  Шардт Максим,  очная форма обучения  курс: 2; группа: ИВТ-1.1 |
| --- | --- |
|  | Научный руководитель:  Профессор Власова Елена Зотиковна |

Санкт-Петербург

2023

**Аппроксимация числа Пи методом Бюффона**

Цель работы: вычислить значение числа Пи с помощью задачи Бюффона о бросании иглы.

**Постановка задачи:** разработать программный модуль для симуляции бросания иглы на плоскость, расчерченную на расстоянии друг от друга параллельными прямыми. Приближенно определить значение Пи по методу Монте-Карло по полученным данным из симуляции.

### *Использованные формулы:*

1. Математическое ожидание того, что отрезок пересекает прямую:   
    где  
   L – длина отрезка,  
   r – расстояние между прямыми.   
   A – расстояние от отрезка до ближайшей прямой,  
    – угол отрезка относительно прямой.
2. Выраженное из (1) значение :

Был создан программный модуль для симуляции бросания иглы на плоскость. После установки начальных значений высоты и ширины экрана, расстояния между прямыми, а также длины и количества бросаемых игл, начинается создание и распределение игл по полю:

| private static List<Needle> CreateNeedles(int width, int height, int stripWidth, double needleLength, int count)  {  var needles = new List<Needle>();  Random random = new Random();  for (int i = 0; i < count; i++)  {  double angle = random.NextDouble() \* 360;  int x = random.Next(width);  int y = random.Next(height);  var start = new PointF(x, y);  var end = new PointF((float)(x + needleLength \* Math.Cos(angle)), (float)(y + needleLength \* Math.Sin(angle)));  needles.Add(new Needle { Start = start, End = end, Intersects = Needle.isIntersecting(start, end, stripWidth, width) });  }  return needles;  } |
| --- |

Далее проводится проверка на пересечение иглы какой либо из прямых:

| public static bool isIntersecting(PointF start, PointF end, int stripWidth, int planeHeight)  {  if (start.X > end.X)  {  PointF temp = start;  start = end;  end = temp;  }   int startStrip = (int)Math.Floor(start.X / stripWidth);  int endStrip = (int)Math.Floor(end.X / stripWidth);   for (int i = startStrip; i <= endStrip; i++)  {  float x = i \* stripWidth;  if (x >= start.X && x <= end.X)  {  return true;  }  }   return false;  } |
| --- |

Иглы, которые пересекают прямые пометим красным цветом, остальные - зеленым. Таким образом получается следующее поле:

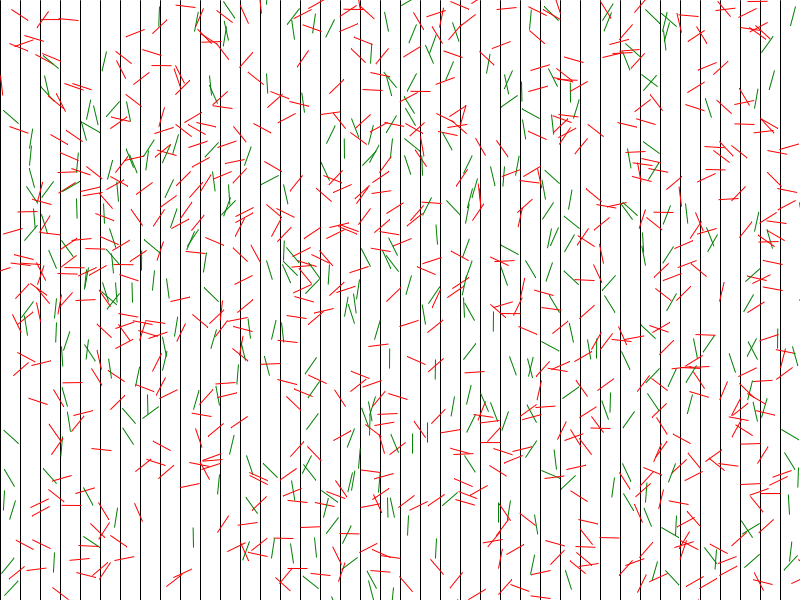


Рис. 1. Поле задачи Бюффона.

По данным симуляции можно составить график отношения попавших иголок к брошенным для каждой иголки:

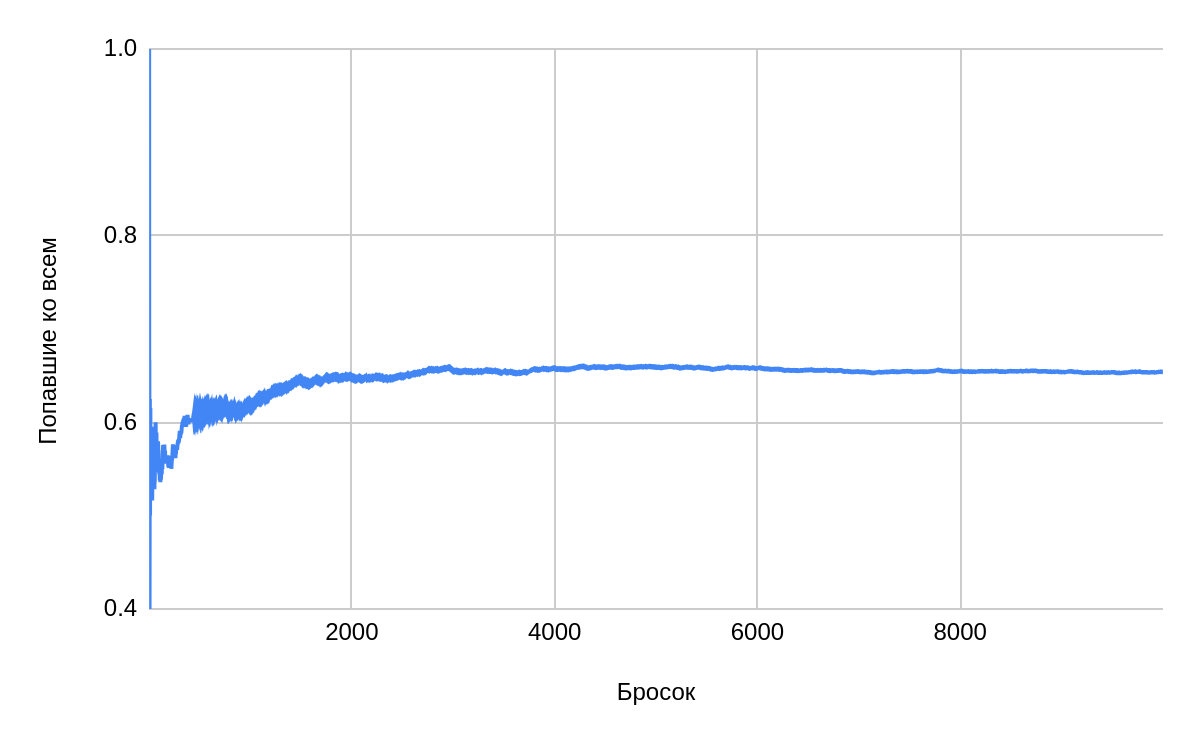


Рис. 2. Отношение попаданий к общему количеству бросков.

Используя данные графика Рис. 2 для вычисления значения числа Пи по формуле (2), возможно составить следующий график:

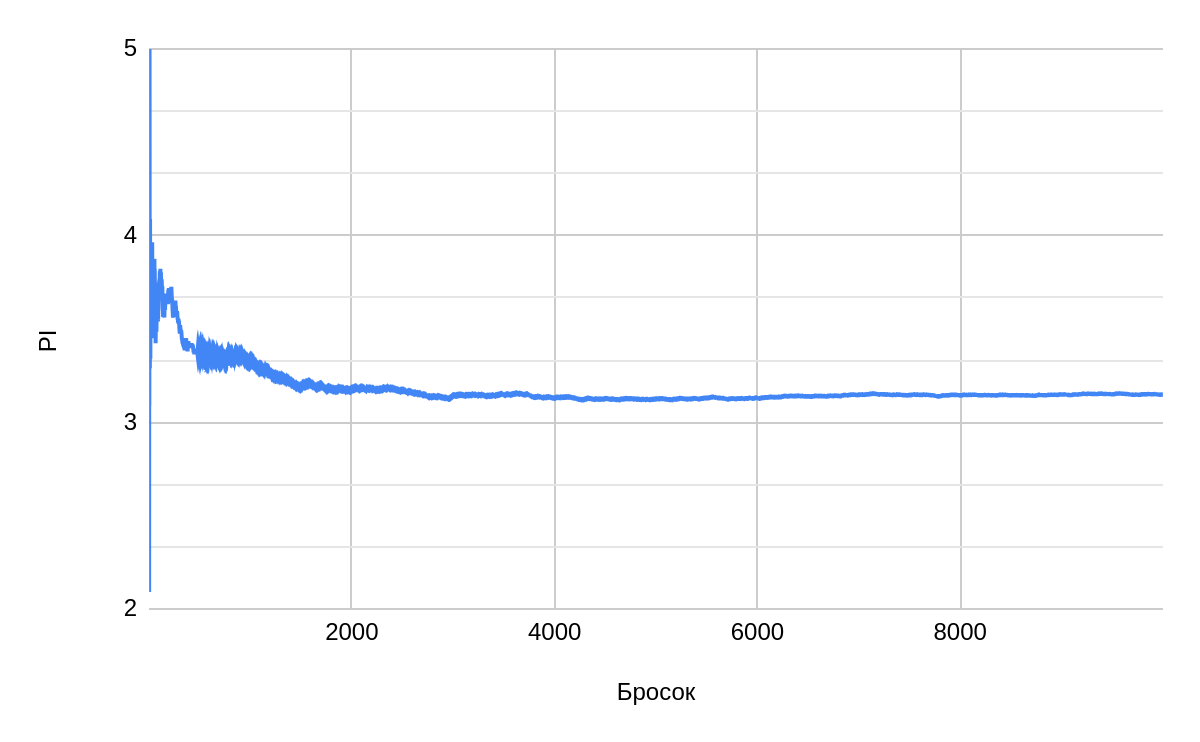


Рис. 3. Вычисленное значение Пи для каждого броска.

Для конечной аппроксимации значения необходимо найти среднее от всех значений по Рис. 3. Так как при низком количестве бросков значение крайне неточно, то будет взята последняя половина значений.

| Количество бросков | Значение | Ошибка |
| --- | --- | --- |
| 1000 | 3.15555 | 0.01395 |
| 10000 | 3.14186 | 0.00027 |
| 100000 | 3.14512 | 0.00353 |

Таблица. 1. Результаты вычислений

Результаты вычислений оказались наиболее точными при значениях расстояния между прямыми и иглы равными 20-ти условным единицам, а также количестве бросков равном 10000. Понижение точности аппроксимации может быть связано как с неточностью модели, так и с неравномерным распределением случайных значений.

**Вывод:** был разработан программный модуль для симуляции бросания иглы на плоскость, расчерченную на расстоянии друг от друга параллельными прямыми. Приближенное значение Пи было определено по методу Монте-Карло по полученным данным из симуляции.

## Литература

1. Math Surprises: An Example [Электронный ресурс]. Архивная копия от 30 января 2012 на Wayback Machine (англ.). Режим доступа: https://web.archive.org/web/20120130001049/http://www.math.utah.edu/~pa/math/surprises.html (дата обращения: 29.06.2023).
2. Hall, A. On an experimental determination of Pi [Текст] // The Messenger of Mathematics. — 1872. — Vol. 2. — P. 113-114.