



А.М. Туйчиев, М. Шукуров
Худжандский государственный университет
имени академика Б. Гафурова

ПРИМЕНЕНИЕ ЧИСЛА КАПРЕКАРА В ЯЗЫКАХ ПРОГРАММИРОВАНИЯ PYTHON и C++

Данная работа посвящена изучению и применению числа Капрекара в программировании на языках Python и C++. Число Капрекара представляет собой интересное явление в числовой теории, которое выявляется через особую математическую операцию, заключающуюся в вычитании из числа, образованного упорядоченными цифрами, меньшего числа, сформированного теми же цифрами, расположенными в обратном порядке. Этот процесс демонстрирует специфический характер числовых последовательностей и открывает уникальные возможности для анализа алгоритмов и вычислительных методов.

Число Капрекара, постоянная Капрекара, решения задачи Капрекара в Python, решения задачи Капрекара в C++.

В современном мире числовые теории и алгоритмы занимают значительное место в области программирования и вычислительной математики. Число Капрекара, названное в честь индийского математика Д.Р. Капрекара, представляет собой одну из математических загадок, привлекающих внимание как исследователей, так и программистов. Его уникальные свойства, заключающиеся в способности определенных чисел достигать особого постоянного значения при итеративных преобразованиях, делают его интересным объектом для математического анализа и применения в различных областях программирования.

В данной статье рассматривается реализация алгоритма числа Капрекара на языках программирования Python и C++, которые являются одними из самых востребованных для разработки числовых и аналитических алгоритмов. Мы анализируем специфику каждого языка и исследуем, как особенности их синтаксиса и структуры позволяют достичь оптимальной реализации данного алгоритма. Разработка кода на различных языках способствует углубленному пониманию нюансов каждого из них и выявляет их сильные и слабые стороны при решении алгоритмических задач.

Цель данной работы – продемонстрировать процесс реализации числа Капрекара на разных языках программирования и сравнить их с точки зрения простоты, эффективности и читаемости кода.

Даттарая Рамчандра Капрекар (англ. Dattaraya Ramchandra Kaprekar) родился 17 января 1905 года в штате Махараштра. Он получил среднее образование в Тейне и окончил Университет Мумбаи в 1929 году. С 1930 по 1962 год работал учителем математики в городе Насик (на западе Индии). Капрекар изучал свойства чисел и в 1949 году открыл одно из свойств числа 6174. Тогда оно будет называться «число Капрекара». [3] В статье мы подчеркиваем о свойствах «числа Капрекара» и создадим программу на Python и

C++. Также упомянем постоянную Капрекара и дополнительные свойства числа 495.

Постоянная Капрекара и его алгоритм

Алгоритм Капрекара заключается в следующем: «для любого четырехзначного числа с неповторяющимися цифрами (в десятичной системе счисления) записываем его цифры сначала в порядке возрастания, а затем в порядке убывания. Затем из большего числа вычитаем меньшее и повторяем процесс до тех пор, пока не получим число 6174. Этот алгоритм всегда достигает результата не более чем за 7 итераций» [2].

При преобразовании двузначных чисел с помощью алгоритма Капрекара получается число 9, тогда смысл алгоритма теряется, для продолжения алгоритма нам нужно число не меньше двузначного.

Пример 1.

Из этих трехзначных чисел: 123; 225; 120 – преобразуйте алгоритм Капрекара.

Для числа 123:

- 1) $321 - 123 = 198$;
- 2) $981 - 189 = 792$;
- 3) $972 - 279 = 693$;
- 4) $963 - 369 = 594$;
- 5) $954 - 459 = 495$.

Для числа 225:

- 1) $522 - 225 = 297$;
- 2) $972 - 279 = 693$;
- 3) $963 - 369 = 594$;
- 4) $954 - 459 = 495$.

Для числа 120:

- 1) $210 - 12 = 198$;
- 2) $981 - 189 = 792$;
- 3) $972 - 279 = 693$;
- 4) $963 - 369 = 594$;
- 5) $954 - 459 = 495$.

Пример 2.

Из этих четырехзначных чисел: 2005; 1887; 1920 – преобразуйте алгоритм Капрекара.

Для числа 2005:

- 1) $5200 - 25 = 5175$;
- 2) $7551 - 1557 = 5994$;
- 3) $9954 - 4599 = 5355$;
- 4) $5553 - 3555 = 1998$;
- 5) $9981 - 1899 = 8082$;
- 6) $8820 - 288 = 8532$;
- 7) $8532 - 2358 = 6174$.

Для числа 1887:

- 1) $8871 - 1788 = 7083$;
- 2) $8730 - 378 = 8352$;
- 3) $8532 - 2358 = 6174$.

Для числа 1920:

- 1) $9120 - 129 = 9081$;
- 2) $9810 - 189 = 9621$;
- 3) $9621 - 1269 = 8352$;
- 4) $8532 - 2358 = 6174$.

До сих пор остается неизвестным, почему натуральные числа подчиняются такому алгоритму. Этот алгоритм может символизировать красоту и изящество математики. В таблице приведены постоянные Капрекара для всех чисел.

Таблица

Постоянные Капрекара

Количество цифр в числе	Постоянная Капрекара
2	Не имеет постоянного
3	495
4	6174
5	Не имеет постоянного
6	549 945, 631 764
7	Не имеет постоянного
8	97 508 421, 63 317 664
9	864 197 532, 554 999 445
10	9 753 086 421, 6 333 176 664, 997 508 420

Способы выражения числа 495

1) $3^2 \cdot 5 \cdot 11 = 495$;

2) $\sum_{k=2}^5 k! + 7^3 = 2! + 3! + 4! + 5! + 7^3 = 2 + 6 + 24 + 120 + 343 = 495$;

3) $\sum_{k=0}^8 2^k - 2^{2^2} = \frac{2^9 - 1}{2 - 1} - 2^4 = 511 - 16 = 495$;

4) $\frac{5^5}{5} - \prod_{k=1}^5 k - \sum_{k=1}^4 k = 5^4 - 5! - \frac{4(4+1)}{2} = 625 - 120 - 10 = 495$;

5) $\sum_{k=2}^4 (2k-1)^3 = 3^3 + 5^3 + 7^3 = 27 + 125 + 343 = 495$;

6) $\sum_{k=1}^{10} k - 21^2 - 1^2 = \frac{10(10+1)}{2} + 441 - 1 = 55 + 440 = 495$;

7) $\sum_{k=1}^3 k^2 + 9^2 + 20^2 = \frac{3 \cdot 4 \cdot 7}{6} + 81 + 400 = 14 + 481 = 495$;

Основные свойства числа 6174

6174 является числом Харшад (великая радость), потому что оно делится на сумму своих чисел:

$$6174 = (6 + 1 + 7 + 4) \cdot 343.$$

Число 6174 известно как число Цумкеллера (англ. Zumkeller number), поскольку множество делителей числа 6174 может иметь два подмножества с одинаковой суммой (7800). Не существует натурального числа, которое при делении на сумму его чисел дает 6174. Ближайшие числа со следующими свойствами: 6123; 6150; 6185; 6189 [3, 4].

Сумма квадратов простых множителей числа 6174 равна квадрату числа 13:

$$2^2 + 3^2 + 3^2 + 7^2 + 7^2 + 7^2 = 4 + 2 \cdot 9 + 3 \cdot 49 = 169 = 13^2.$$

Число 6174 можно записать как сумму первых трех степеней 18:

$$\sum_{k=1}^3 18^k = 18^1 + 18^2 + 18^3 = 18 + 324 + 5832 = 6174.$$

Решения задачи Капрекара в Python

```
n=int(input("n=")) # Любое целое число
w=1 # алгоритм счетчика
while n!=6174: # алгоритм продолжается до того
    времени по n не станет 6174
    a=[]
    h=n//1000
```

Делит число n на цифры и образует массив:

```
a.append(h)
s=n//100%10
a.append(s)
d=n//10%10
a.append(d)
v=n%10
a.append(v)
```

Создает минимальное число:

```
a.sort()
k=1000
minimum=0
maximum=0
for i in a:
    minimum+=i*k
    k/=10
```

Создает максимальное число

```
a.reverse()
k=1000
for i in a:
    maximum+=i*k
    k/=10
```

Разница между максимальным и минимальным числом:

```
n=maximum-minimum
print(f" {int(w)}")
{int(maximum)}-
-{int(minimum)}=
={int(n)}")
```

Увеличивает значение счетчика:
выводит результат
w+=1

Решения задачи Капрекара в C++

```
#include <bits/stdc++.h>
using namespace std;
int main()
{
Создание массива
int n,t=1;
vector <int> a;
cout << "n=";
cin >> n;
while (n!=6174)
Алгоритм продолжается до того времени по n не ста-
нет int h=n/1000
//
```

Дополняет массив:

```
a.push_back(h);
int s=n/100%10;
a.push_back(s);
int d=n/10%10
a.push_back(d);
int v=n%10;
a.push_back(v);
```

Создает минимальное число

```
sort(a.begin(), a.end());
int k=1000;
int minimum=0;
for(int i=0; i<a.size(); ++i
minimum+=a[i]*k;
k/=10;
```

Создает максимальное число

```
k=1000;
```

```
int maximum=0;
reverse(a.begin(), a.end());
for(int i=0; i<a.size(); ++i)
{
maximum+=a[i]*k;
k/=10;
}
```

Очищает массив:
a.clear();
n=maximum-minimum;

Разница между максимальным и минимальным чис-
лом:
cout << t << ") " << maximum << "-" << minimum <<
"=" << n << endl;

Значение счетчика увеличивается на единицу:

```
t++; //
}
return 0;
```

Полученные результаты и выводы могут спо-
собствовать более осознанному выбору языковых ин-
струментов для решения математических задач и раз-
работки эффективных алгоритмов, что актуально как
для обучающихся программированию, так и для ис-
следователей в области прикладной математики и
информатики.

Литература

- Исаев, А. В. Параллельные миры II или струк-
тура пространства-времени / А. В. Исаев. – Санкт-
Петербург : «ЛИСС», 2002. – 240 с.
- Казанджиев, П. Исследования, свързани с Кон-
стантата на Капрекар / П. Казанджиев // Vocational
Education. – 2018. – Т. 20, № 3.
- Корпал, Г. Празднование 110-летия со дня рож-
дения доктора Капрекара / Г. Корпал // Конспект
лекции доступен по ссылке. – 2015. – URL: [https://
gaurish4math.files.wordpress.com/2014/12/kaprekar.pdf](https://gaurish4math.files.wordpress.com/2014/12/kaprekar.pdf)
(дата обращения: 05.03.2025). – Текст : электронный.
- Толпыго, А. К. Числа харшад / А. К. Толпыго // *Квант*. – 2020. – №. 10. – С. 34–38.

A.M. Tychiev, M. Shukurov

Khujaud State University named after Academician B. Gafurov

APPLICATION OF CAPRECAR NUMBER IN PYTHON AND C++ PROGRAMMING LANGUAGES

This work is devoted to the study and application of the Kaprekar number in programming in Python and C++. The Kaprekar number is an interesting phenomenon in number theory, which is revealed through a special mathematical operation that consists of subtracting a number formed by ordered digits from a smaller number formed by the same digits arranged in reverse order. The Kaprekar process demonstrates the specific nature of numerical sequences and opens up unique opportunities for the analysis of algorithms and computational methods.

Kaprekar number, Kaprekar constant, solutions to the Kaprekar problem in Python, solutions to the Kaprekar prob-
lem in C++.