

## РЕШЕНИЕ ПРОБЛЕМЫ КАНТЕЛЛИ ДЛЯ ТАБЛИЧНЫХ ФУНКЦИИ

### **Иванченко Владислав Александрович**

студент, Белорусский государственный университет информатики и радиоэлектроники, Беларусь, г. Минск

### **Якуба Владислав Витальевич**

студент, Белорусский государственный университет информатики и радиоэлектроники, Беларусь, г. Минск

### **Баженова Ирина Владимировна**

научный руководитель, канд. физ.-мат. наук, доц., Белорусский государственный университет информатики и радиоэлектроники, Беларусь, г. Минск

## KANTELLI`S PROBLEM SOLUTION FOR TABULAR FUNCTIONS

### ***Vladislav Ivanchenko***

*Student, Belorussian state university informatics and radioelectronics, Belarus, Minsk*

### ***Vladislav Yakuba***

*Student, Belorussian state university informatics and radioelectronics, Belarus, Minsk*

### ***Irina Bazhenova***

*Scientific director, Ph. D. phys.-math. sciences, associate professor, Belorussian state university informatics and radioelectronics, Belarus, Minsk*

**Аннотация.** Теория вероятностей является чрезвычайно важным разделом не только математики, но и важным математическим аппаратом в радиотехнике, связи, ядерной, квантовой и молекулярной физике, метрологии и многих других дисциплинах. В данной статье показаны результаты решения проблемы Кантелли для табличных функции, с использованием программирования на языке C++. В результате решения было выявлено подтверждение гипотезы для табличных функции.

**Abstract.** Probability theory is an extremely important section not only of mathematics, but also an important mathematical apparatus in radio engineering, communications, nuclear, quantum and molecular physics, metrology and many other disciplines. This article shows the results of solving Cantelli's problem for table functions using C++ programming. As a result of the solution, confirmation of the hypothesis for tabular functions was revealed.

**Ключевые слова:** теория вероятностей, математическая статистика, высшая математика,

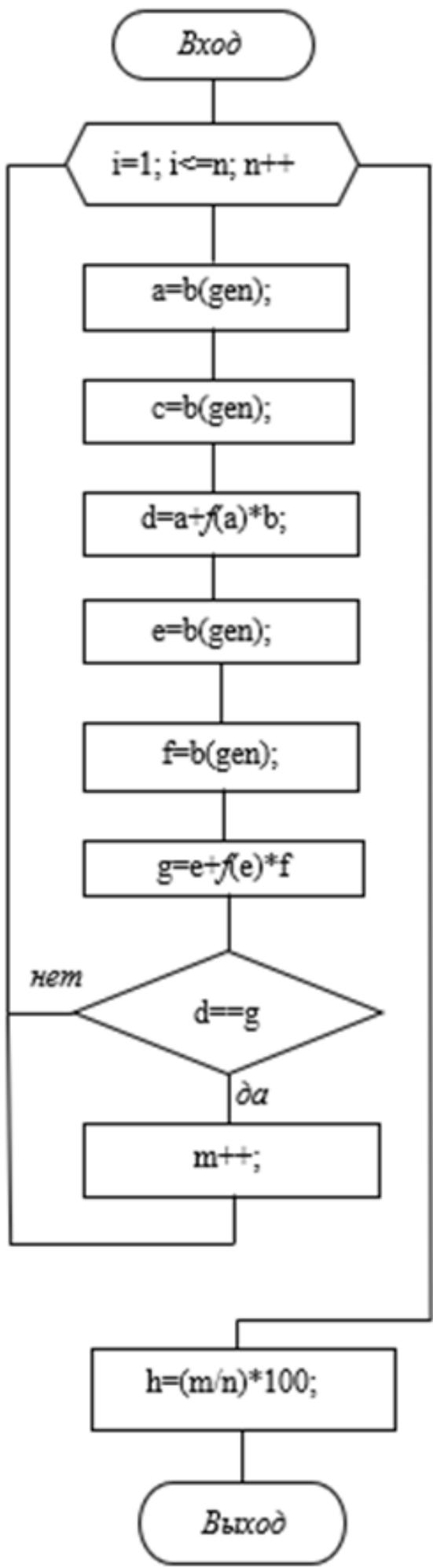
общая физика, молекулярная физика, квантовая физика, статистическая физика, ядерная физика, физика элементарных частиц, радиотехника, связь, метрология, проблема Кантелли.

**Keywords:** probability theory, math statistics, higher mathematics, common physics, molecular physics, quantum physics, statistic physics, nuclear physics, particle physics, radio engineering, communications, metrology, Cantelli`s problem.

Проблема Кантелли является одной из открытых проблем в теории вероятностей. Сама проблема представляет из себя следующие условия: есть две случайные независимые друг от друга величины  $X$  и  $Y$ , они подчиняются нормальному распределению  $N(0;1)$ ,  $f(x)$  является измеримой неотрицательной функцией, а выражение  $X+f(X)*Y$  тоже имеет нормальное распределение. Является ли функция почти всюду однозначной? [1]

В статье представлен вариант решения этой проблемы, написанный соавторами на языке C++. Блок-схема решения для одной функции показана на рисунке 1. Так как функции должны быть измеримыми и неотрицательными, то не все табличные функции подходят для такого исследования. В данной статье рассматриваются следующие функции:  $x^2$ ,  $2^x$ ,  $\log_2(x)$ ,

$\ln(x)$ ,  $e^x$ ,  $|x|$ ,  $x!$ ,  $\sqrt{x}$ .



## *Рисунок 1. Блок-схема цикла для нахождения решения задачи*

Сама программа выглядит так:

```
#include <iostream>
#include <math.h>
#include <ctype.h>
#include <random>
#include <complex>
#include <conio.h>
using namespace std;
int factorial(int cc) {
    if (cc < 2) return 1;
    return cc * factorial(cc - 1);    //Вычисление факториала для функции x!
}
int main() {
    long double a, b, c, f, g, h, i, n = 0, j, k, l, m, r, t, o, p, q = 0, w, s, u, y, x, z, aa, ab, ac = 0, ad, ae,
    af, an = 0, ao, ap, aq, ar, as, at, au, av = 0, aw, ax, ay, az, ba, bb, bc, bd, be, bf = 0, bg, bh, bi, bj,
    bk, bl, bm, bo, bp, bq=0, br, bs, bt, bw, bz=0, ca=0, cb;
    complex <long double> am, al;
    long double const E =
```

2.7182818284590452353602874713526624977572470936999595749669676277240  
76630353547594571382178525166427427466391932003059921817413596629043  
57290033429526059563073813232862794349076323382988075319525101901157  
38341879307021540891499348841675092447614606680822648001684774118537  
42345442437107539077744992069551702761838606261331384583000752044933  
82656029760673711320070932870912744374704723069697720931014169283681  
90255151086574637721112523897844250569536967707854499699679468644549  
05987931636889230098793127736178215424999229576351482208269895193668  
03318252886939849646510582093923982948879332036250944311730123819706  
84161403970198376793206832823764648042953118023287825098194558153017  
56717361332069811250996181881593041690351598888519345807273866738589  
42287922849989208680582574927961048419844436346324496848756023362482  
70419786232090021609902353043699418491463140934317381436405462531520  
96183690888707016768396424378140592714563549061303107208510383750510  
1157477041718986106873969655212671546889570350354; //Первые 1000 знаков

## числа Эйлера

```
int bu, bv, bx, by;

for (h = 1; h <= 1000; h++) {           //Проводящий опыт, цикл, где число в неравенстве -
    количество итерации

        random_device rd;

        mt19937 gen(rd());

        normal_distribution<>d(0.0, 1.0);

        long double a = d(gen);

        long double b = d(gen);         //Генерация случайных величин

        c = a + pow(a, 2) * b;          //a+f(a)*b

        normal_distribution<>c(0.0, 1.0);

        long double f = d(gen);

        long double g = d(gen);

        i = f + pow(f, 2) * g;

        normal_distribution<>i(0.0, 1.0);
```

```

    if (c == i)n++;           //Проверка на однозначность в данном опыте
}
j = (n / h) * 100;          //Доля события, где функция однозначна
cout << "x^2 is unambiguous on - " << j << "%" << endl;    //Вывод результатов
for (k = 1; k <= 1000; k++) {
    random_device pp;
    mt19937 gen(pp());
    normal_distribution<>d(0.0, 1.0);
    long double l = d(gen);
    long double m = d(gen);
    r = l + fabs(l) * m;
    normal_distribution<>r(0.0, 1.0);
    long double t = d(gen);
    long double o = d(gen);
    p = t + fabs(t) * o;
    normal_distribution<>p(0.0, 1.0);
    if (r == p)q++;
}
w = (q / k) * 100;
cout << "|x| is unambiguous on - " << w << "%" << endl;
for (s = 1; s <= 1000; s++) {
    random_device ut;
    mt19937 gen(ut());
    normal_distribution<>d(0.0, 1.0);
    long double u = d(gen);
    long double y = d(gen);
    aa = u + pow(E, u) * y;
    normal_distribution<>aa(0.0, 1.0);
    long double x = d(gen);
    long double z = d(gen);
}

```

```

    ab = x + pow(E, x) * z;
    normal_distribution<>ab(0.0, 1.0);
    if (aa == ab)ac++;
}
ad = (ac / s) * 100;
cout << "e^x is unambiguous on - " << ad << "%" << endl;
for (ae = 1; ae <= 1000; ae++) {
    random_device af;
    mt19937 gen(af());
    normal_distribution<>d(0.0, 1.0);
    complex <long double> ag = d(gen);
    complex <long double> ai = d(gen);
    am = ag + sqrt(ag) * ai;
    normal_distribution<>am(0.0, 1.0);
    complex <long double> aj = d(gen);
    complex <long double> ak = d(gen);
    al = aj + sqrt(aj) * ak;
    normal_distribution<>al(0.0, 1.0);
    if (al == am)an++;
}
ao = (an / ae) * 100;
cout << "x^(1/2) is unambiguous on - " << ao << "%" << endl;
for (j = 1; j <= 1000; j++) {
    random_device k;
    mt19937 gen(k());
    normal_distribution<>d(0.0, 1.0);
    long double ap = d(gen);
    long double aq = d(gen);
    ar = ap + log(ap) * aq;
    normal_distribution<>ar(0.0, 1.0);
}

```

```

    long double as = d(gen);
    long double at = d(gen);
    au = as + log(as) * at;
    normal_distribution<>au(0.0, 1.0);
    if (ar == au)av++;
}
aw = (av / j) * 100;
cout << "ln(x) is unambiguous on - " << aw << "%" << endl;
for (ax = 1; ax <= 1000; ax++) {
    random_device ay;
    mt19937 gen(ay());
    normal_distribution<>d(0.0, 1.0);
    long double az = d(gen);
    long double ba = d(gen);
    bb = az + pow(2, ba) * ba;
    normal_distribution<>bb(0.0, 1.0);
    long double bc = d(gen);
    long double bd = d(gen);
    be = bc + pow(2, bc) * bd;
    normal_distribution<>be(0.0, 1.0);
    if (bb == be)bf++;
}
bg = (bf / ax) * 100;
cout << "2^x is unambiguous on - " << bg << "%" << endl;
for (bh = 1; bh <= 1000; bh++) {
    random_device bi;
    mt19937 gen(bi());
    normal_distribution<>d(0.0, 1.0);
    long double bj = d(gen);
    long double bk = d(gen);

```

```

    bl = bj + (log(bj)/log(2)) * bk;
    normal_distribution<>bl(0.0, 1.0);
    long double bm = d(gen);
    long double bo = d(gen);
    bp = bm + (log(bm)/log(2)) * bo;
    normal_distribution<>bp(0.0, 1.0);
    if (bl == bp)bq++;
}
br = (bq / bh) * 100;
cout << "log2(x) is unambiguous on - " << br << "%" << endl;
for (bs = 1; bs <= 1000; bs++) {
    random_device bt;
    mt19937 gen(bt());
    normal_distribution<>d(0.0, 1.0);
    int bu = d(gen);
    int bv = d(gen);
    bw = bu + factorial(bu) * bv;
    normal_distribution<>bw(0.0, 1.0);
    int bx = d(gen);
    int by = d(gen);
    bz = bx + factorial(bx) * by;
    normal_distribution<>bz(0.0, 1.0);
    if (bw == bz)ca++;
}
cb = (ca / bs) * 100;
cout << "x! is unambiguous on - " << cb << "%" << endl;
return 0;
}

```

Результаты работы программы представлены на рисунках 2 - 5:

```
x^2 is unambiguous on - 90.9091%
|x| is unambiguous on - 90.9091%
e^x is unambiguous on - 90.9091%
x^(1/2) is unambiguous on - 90.9091%
ln(x) is unambiguous on - 90.9091%
2^x is unambiguous on - 90.9091%
log2(x) is unambiguous on - 90.9091%
x! is unambiguous on - 90.9091%
```

*Рисунок 2. Результаты работы программы при 10 итерациях*

```
x^2 is unambiguous on - 99.0099%
|x| is unambiguous on - 99.0099%
e^x is unambiguous on - 99.0099%
x^(1/2) is unambiguous on - 99.0099%
ln(x) is unambiguous on - 99.0099%
2^x is unambiguous on - 99.0099%
log2(x) is unambiguous on - 99.0099%
x! is unambiguous on - 99.0099%
```

*Рисунок 3. Результаты работы программы при 100 итерациях*

```
x^2 is unambiguous on - 99.9001%
|x| is unambiguous on - 99.9001%
e^x is unambiguous on - 99.9001%
x^(1/2) is unambiguous on - 99.9001%
ln(x) is unambiguous on - 99.9001%
2^x is unambiguous on - 99.9001%
log2(x) is unambiguous on - 99.9001%
x! is unambiguous on - 99.9001%
```

*Рисунок 4. Результаты работы программы при 1000 итерациях*

```
x^2 is unambiguous on - 99.99%
|x| is unambiguous on - 99.99%
e^x is unambiguous on - 99.99%
x^(1/2) is unambiguous on - 99.99%
ln(x) is unambiguous on - 99.99%
2^x is unambiguous on - 99.99%
log2(x) is unambiguous on - 99.99%
x! is unambiguous on - 99.99%
```

*Рисунок 5. Результаты работы программы при 10000 итерациях*

В результате можно сказать, что, при вышеперечисленных условиях, функции, по крайней мере те, которые рассматриваются в данной статье, почти всюду однозначны. Это может использоваться в математике, статистике, физике и других дисциплинах как математический аппарат.

#### **Список литературы:**

1. Дороговцев, А. Я. Математика сегодня / А. Я. Дороговцев. – Киев. : Вища школа, 1983. – 192с.