УДК 004

Реализация алгоритма арифметического кодирования на языке C++

Ленкин Алексей Викторович
Приамурский государственный университет имени Шолом-Алейхема
студент

Аннотация
В данной статье рассмотрен алгоритм арифметического шифрования. Описана реализация данного алгоритма с помощью языка программирования C++.
Ключевые слова: C++, шифрование, арифметическое кодирование

Implementation of the algorithm of arithmetic coding in C++ language

Lenkin Aleksei Viktorovich
Sholom-Aleichem Priamursky State University
student

Abstract
This article describes the algorithm of arithmetic encryption. The implementation of this algorithm with the help of C++ programming language is described.
Keywords: C++, encryption, arithmetic coding

Научный руководитель:
Лучанинов Дмитрий Васильевич
Приамурский государственный университет имени Шолом-Алейхема
старший преподаватель кафедры информационных систем, математики и методик преподавания

Сейчас существует множество алгоритмов сжатия информации. Большинство из них широко известны, но есть и некоторые весьма эффективные, но, тем не менее, малоизвестные алгоритмы [1].

Одним из таких является арифметическое кодирование. Основой арифметического кодирования является неопубликованный алгоритм П. Элайеса. Также существенный вклад в развитие данного метода в разное время внесли П. Говард, М. Гуаззо, Д. Клири, Г. Лэнгдон, Э. Моффат, Р. Нил, Р. Паско, Д. Риссанен, Ф. Рубин, Я. УиттениМ. Шиндлер[2].

Целью исследования является анализ алгоритма арифметического кодирования и его реализация на языке C++.

Исследованиями в данной теме занимались следующие авторы. Самохвалов А.В. показал «Комплексное использование алгоритма кодирования длин серий и арифметического кодирования для сжатия изображений»[3]. Фионов А.Н. описал «Методы эффективной рандомизации

сообщений, базирующиеся на омофонном и арифметическом кодировании»[4]. «Адаптивное арифметическое кодирование в стандарте jpeg 2000» было изучено исследователями Беляев Е.А., Тюрликов А.М., Уханова А.С.[5].

Предположим, что в используемом алфавите N символов a_1,…,a_N, с частотами p_1,…,p_N, соответственно. Тогда алгоритм арифметического кодирования будет выглядеть следующим образом[6]:

1. В качестве рабочего полуинтервала взять [0;1);
2. Разбить рабочий полуинтервал на N непересекающихся полуинтервалов. При этом длина i-ого полуинтервала пропорциональна p_i.
3. Если не достигнут конец сообщения, в качестве нового рабочего интервала выбрать i-ый полуинтервал и перейти к шагу 2. В противном случае, вернуть любое число из рабочего полуинтервала.

Приведём пример:

1. Возьмём слово «abaac».
2. Определим алфавит и частоту появления символов в слове.
3. Определим интервалы всего алфавита по частоте появления на полуинтервале [0; 1).
4. Начиная с первой буквы, будем перераспределять интервалы.
5. Любое число из интервала последнего символа сообщения будет являться шифром.

На рисунке 1 представлен процесс кодирования сообщения «abaac».

Реализуем данный алгоритм на языке C++. Разработанная программа работает по следующему алгоритму:

1. У введенного сообщения формируется алфавит, происходит подсчет частоты встречи символов и сортировка по ней всего алфавита
2. Определения начальных интервалов для символов алфавита
3. Шифрование сообщения
4. Вывод зашифрованной последовательности

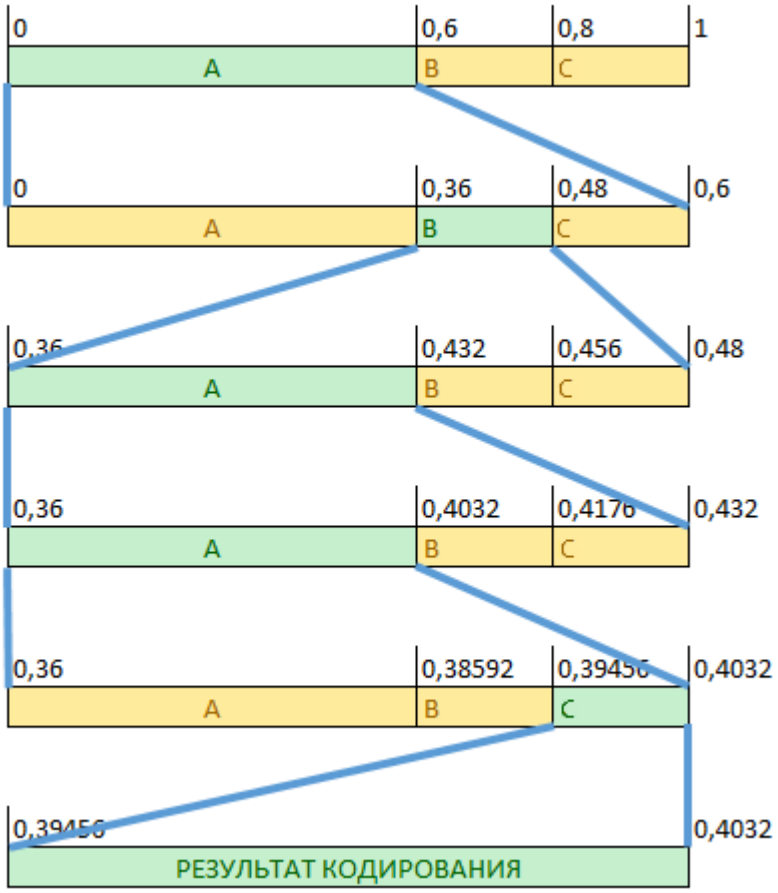

Рисунок 1. Арифметическое шифрование сообщения «abaac»

Листинг представлен ниже:

```
#include <iostream>
#include <algorithm>
#include <string>
#include <windows.h>
using namespace std;
int main()
{
	string s;
	setlocale(LC_ALL, "Russian");
	SetConsoleCP(1251);
	SetConsoleOutputCP(1251);
	int c[255]={0};
	int k,t=0;
	cout<<"Введите сообщение:"<<endl;
	getline(cin,s);
	//Формирование алфавита и его сортировка
	for (int i=0;i<s.length();i++)
	{
		c[(int)s[i]]++;
	}
	for (int i=0;i<255;i++)
	{
		if (c[i]!=0) k++;
```

```
     }
     char alp[k];
     char temp;
     int count[k];
     k=0;
     for (int i=0;i<255;i++)
     {
          if (c[i]!=0)
          {
               alp[k]=(char)i;
               count[k]=c[i];
               k++;
          }
     }
     for (int i = 0; i < k; i++)
     {
          for (int j = k - 1; j > i; j--)
          {
               if (count[j] > count[j - 1])
               {
                swap (count[j], count[j - 1]);
                swap (alp[j], alp[j - 1]);
               }
          }
     }
     double inter[k][2]; //Определение начального интервала
     double down,up;
     inter[0][0]=0.0;
     inter[0][1]=(double)count[0]/s.length();
     for (int i=1;i<k;i++)
     {
          inter[i][0]=inter[i-1][1];
          inter[i][1]=inter[i-1][1]+(double)count[i]/s.length();
     }
          //Шифрование сообщения
     for (int i=0;i<s.length()-1;i++)
     {
          for (int j=0;j<k;j++)
          {
               if (s[i]==alp[j])
               {
                    down=inter[j][0];
                    up=inter[j][1];
                    inter[0][0]=down;
                    inter[0][1]=down+(up-
down)*count[0]/s.length();
                    for (int l=1;l<k;l++)
                    {
                         inter[l][0]=inter[l-1][1];
                         inter[l][1]=inter[l-1][1]+(up-
down)*count[l]/s.length();
                    }
               }
```

```
            }
      }

      for (int j=0;j<k;j++)
            {
                  if (alp[j]==s[s.length()-1])
                  cout <<"Зашифрованное сообщение, любое число в
полуинтервале: ["<< inter[j][0] <<", "<< inter[j][1]<<")";
            }
}
```

Попробуем с помощью программы зашифровать фразу из примера «abaac». Результат на рисунке 2.

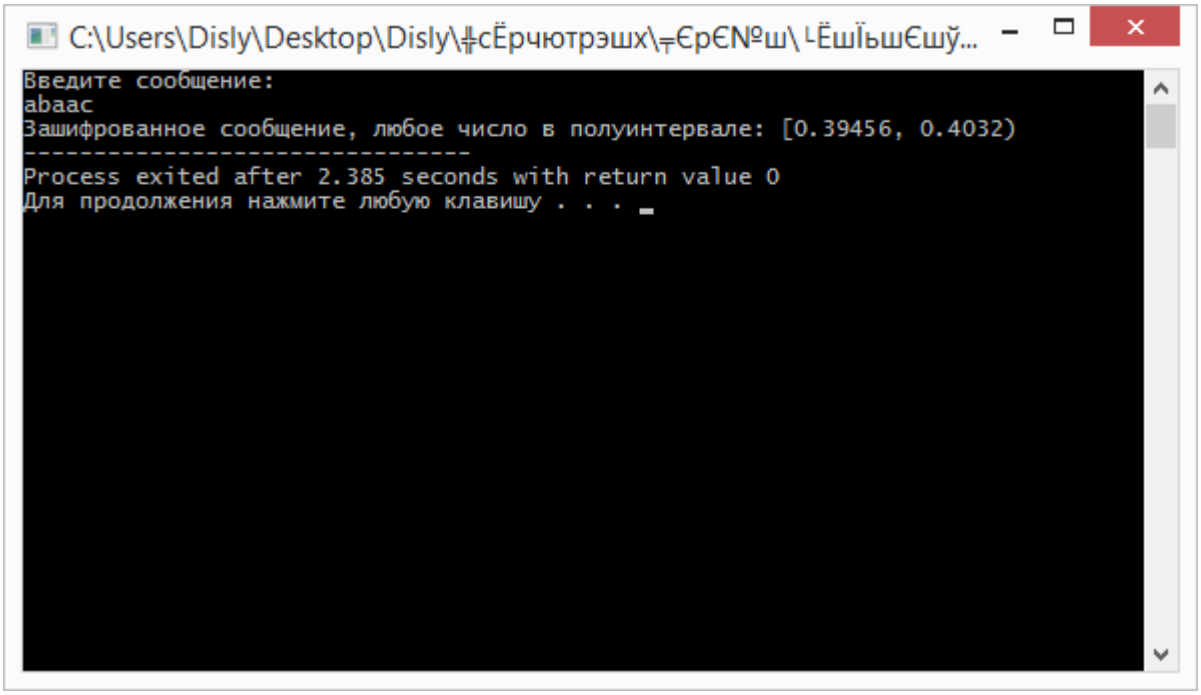

Рисунок 2. Программа для арифметического шифрования

По результату видно, что полученный шифр не отличается от такового в примере.

Таким образом, можно сказать, что арифметическое шифрование является очень эффективным, так как позволяет зашифровать сообщение с энтропией меньше 1 бита на символ. Но процесс кодирования и декодирования является слишком сложным, чтобы использовать этот алгоритм для повседневных задач

Библиографический список

1. Арифметическое кодирование [Электронный ресурс] URL: https://habrahabr.ru/post/130531/ (дата обращения 04.02.2018)
2. Иринёв А., Каширин В. Арифметическое кодирование. // Computer technologies department, ITMO University [Электронный ресурс] URL:

http://rain.ifmo.ru/cat/data/theory/data-compression/arithmetic-coding-2006/article.pdf/ (дата обращения 04.02.2018)

3. Самохвалов А.В. Комплексное использование алгоритма кодирования длин серий и арифметического кодирования для сжатия изображений // Труды международного симпозиума Надежность и качество. 2008. Т. 1. С. 237-240.
4. Фионов А.Н. Методы эффективной рандомизации сообщений, базирующиеся на омофонном и арифметическом кодировании // автореферат диссертации на соискание ученой степени кандидата технических наук. Новосибирск, 1998
5. Беляев Е.А., Тюрликов А.М., Уханова А.С. Адаптивное арифметическое кодирование в стандарте jpeg 2000 // Информационно-управляющие системы. 2007. № 6. С. 28-33.
6. Методы сжатия данных [Электронный ресурс] URL: https://habrahabr.ru/post/251295/ (дата обращения 04.02.2018)