МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

**Отчет по лабораторной работе №8**

**СЖАТИЕ/РАСПАКОВКА ДАННЫХ МЕТОДОМ БАРРОУЗА − УИЛЕРА**

Выполнил:

Cтудент 3 курса 1 группы

Парибок И. А.

Вариант 5

Минск 2022

**Цель**: приобретение практических навыков использования метода Барроуза − Уилера для сжатия/распаковки данных.

**Практическое задание:**

1. Разработать авторское приложение в соответствии с целью лабораторной работы. Входной блок данных может иметь произвольную длину.  
 2. С помощью приложения выполнить прямое и обратное преобразования 3 отдельных блоков данных, состоящих:

а) из собственного имени (можно краткий вариант записи);

б) собственной фамилии;

в) варианта в соответствии с таблицей ниже.

Можно использовать любой из известных методов сортировки символов массива.

Выполнить качественный сравнительный анализ длительности процессов прямого и обратного преобразований в зависимости от длины блока данных.



Рисунок 1 – ­Варианты заданий

3. Перевести первые 3 символа из блока данных, указанного в варианте таблицы, в бинарную последовательность в соответствии с кодами ASCII. Выполнить прямое и обратное преобразование. Оценить время прямого и обратного преобразований.

4. Результаты оформить в виде отчета по установленным правилам.

**Выполнение работы:**

Результаты выполнения задания представлены в рисунках 2-4. Листинг программы изложен в приложении.

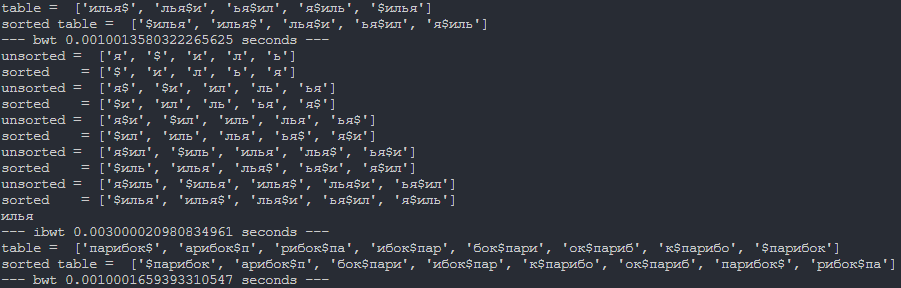


Рисунок 2 – Сжатие и распаковка имени методом Барроуза - Уиллера

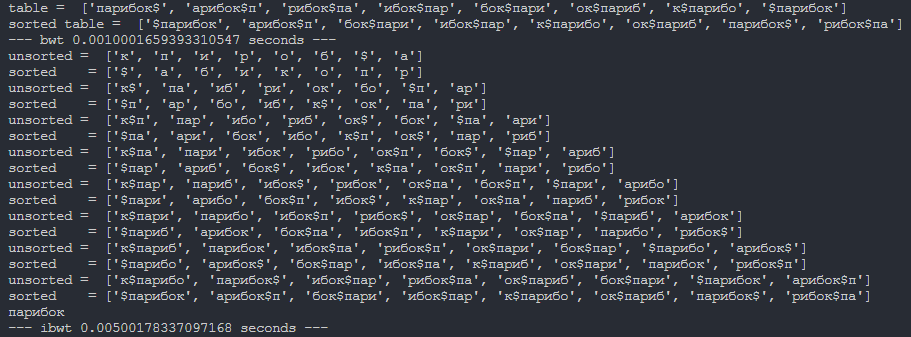


Рисунок 3 – Сжатие и распаковка фамилии методом Барроуза - Уиллера

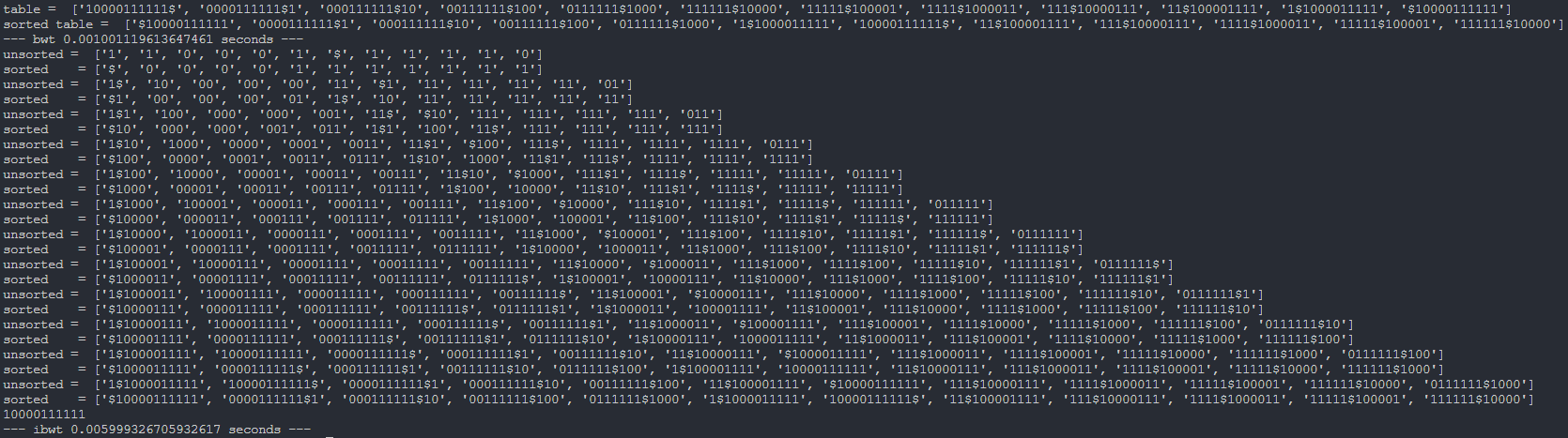


Рисунок 4 – Пример сжатие и распаковка первого символа слова 5-го варианта в двоичном представлении кода ASCII методом Барроуза – Уиллера

**Вывод**: в результате данной лабораторной работы было разработано приложение для с использованием метода Барроуза−Уилера для сжатия/распаковки данных. В ходе сравнения времени выполнения функций сжатия и распаковки, было выявлено что процесс сжатия данных методом Барроуза−Уилера происходит в несколько раз быстрее чем процесс распаковки.

Приложение

import time

def bwt(Input, start\_time):

    assert "$" not in Input                     *# Input string cannot contain $*

    Input = Input + "$"                         *# Add "$" to the end of the string*

    table = [Input[i:] + Input[:i] for i in range(len(Input))]  *# Table of rotations of string*

    print('table = ', table)

    table = sorted(table)

    print('sorted table = ', table)

    last\_column = [row[-1:] for row in table]             *# Last characters of each row*

    bwt = ''.join(last\_column)

    print("--- bwt %s seconds ---" % (time.time() - start\_time))

    start\_time = time.time()

    ibwt(bwt,start\_time)

def ibwt(bwt, start\_time):

    table = [""] \* len(bwt)  *# Make empty table*

    for i in range(len(bwt)):

        table = [bwt[i] + table[i] for i in range(len(bwt))]  *# Add a column of r*

        print('unsorted = ', table)

        table = sorted(table)

        print('sorted    =', table)

    inverse\_bwt = [row for row in table if row.endswith("$")][0]  *# Find the correct row (ending in $)*

    inverse\_bwt = inverse\_bwt.rstrip("$")  *# Get rid of start and end markers*

    print(inverse\_bwt)

    print("--- ibwt %s seconds ---" % (time.time() - start\_time))

variant\_message = 'песнетворчество'[0:1]

ascii\_codes = ''

for ch in  variant\_message:

    ascii\_codes +=  bin(ord(ch)).lstrip("0b")

list\_messages = ['илья', 'парибок', ascii\_codes]

for mes in list\_messages:

    start\_time = time.time()

    bwt(mes,start\_time)