FEDERAL STATE AUTONOMOUS EDUCATIONAL INSTITUTION

OF HIGHER EDUCATION

ITMO UNIVERSITY

Report

on the practical task No. 1

“Experimental time complexity analysis”

Performed by

Alexandra Matveeva

J4134c

Accepted by

Dr Petr Chunaev

St. Petersburg

2021

**Goal**

Experimental study of the time complexity of different algorithms

**Problems**

For each n from 1 to 2000, measure the average computer execution time (using timestamps) of programs implementing the algorithms and functions below for five runs. Plot the data obtained showing the average execution time as a function of n. Conduct the theoretical analysis of the time complexity of the algorithms in question and compare the empirical and theoretical time complexities.

**I.** Generate an n-dimensional random vector with non-negative elements. For , implement the following calculations and algorithms:

1. (constant function);
2. (the sum of elements);
3. (the product of elements);
4. supposing that the elements of are the coefficients of a polynomial 𝑃 of degree *n − 1*, calculate the value 𝑃(1.5) by a direct calculation of   
    (i.e. evaluating each term one by one) and by Horner’s method by representing the polynomial as ;
5. Bubble Sort of the elements of ;
6. Quick Sort of the elements of ;
7. Timsort of the elements of .

**II.** Generate random matrices 𝐴 and 𝐵 of size 𝑛 × 𝑛 with non-negative elements.

Find the usual matrix product for 𝐴 and 𝐵.

**III.** Describe the data structures and design techniques used within the algorithms.

**Brief theoretical part**

An **algorithm** is any well-defined computational procedure that takes some value, or set of values, as input and produces some value, or set of values, as output. An algorithm is thus a sequence of computational steps that transform the input into the output.

**Time complexity** of an algorithm signifies the total time required by the program to run till its completion. It is most commonly expressed using the Big-O notation.

**Constant function** , where is a function that has the same output value (no matter what input value is). Algorithm of calculation of constant function has constant time complexity .

**The sum of elements**

**The product of elements**

**Polynomial function**

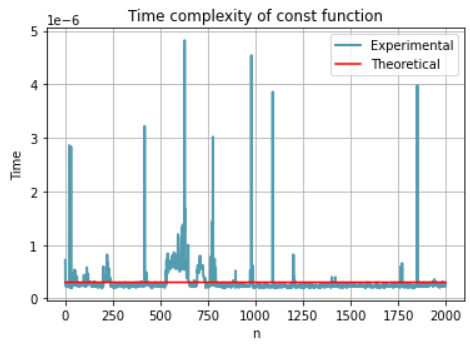
**Horner method**

**Bubble sort**

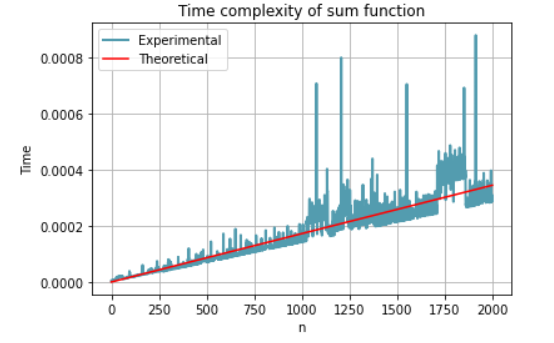
**Quick sort**

**Timsort**

**Results**

****

Pic. 1 – Time complexity of the constant function



Pic. 2 - Time complexity of the sum elements function

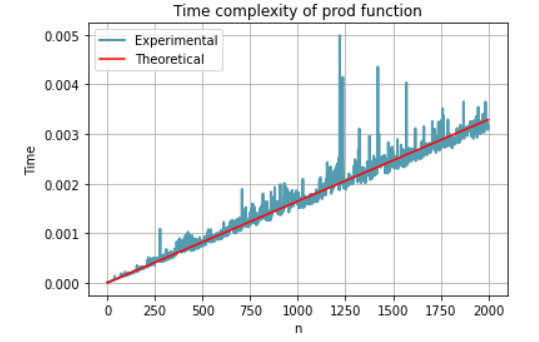


Fig. 3 – Time complexity of the product of the elements

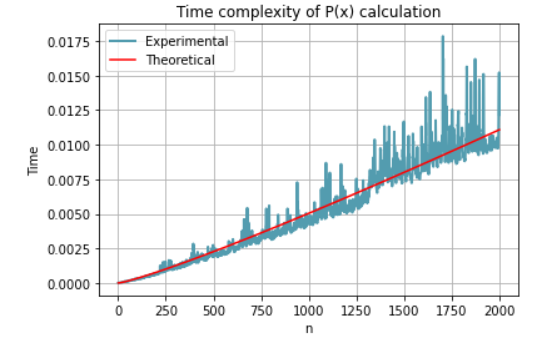


Fig. 4 – Time complexity of the Native method

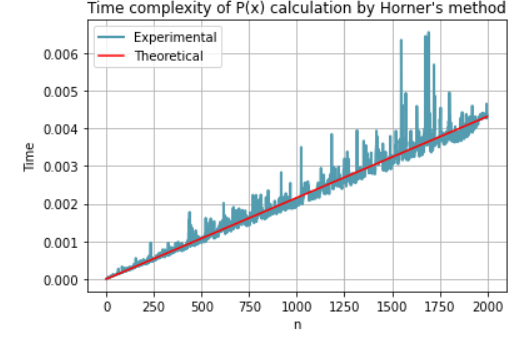


Fig. 5 – Time complexity of Horner’s method

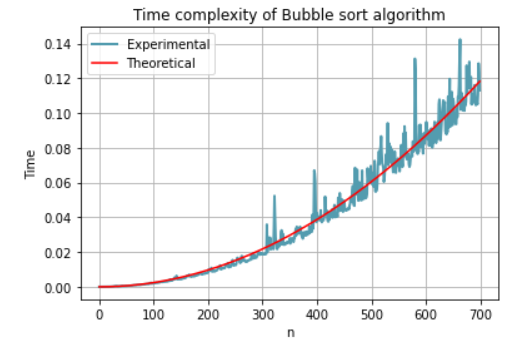


Fig. 6 – Time complexity of Bubble Sort algorithm

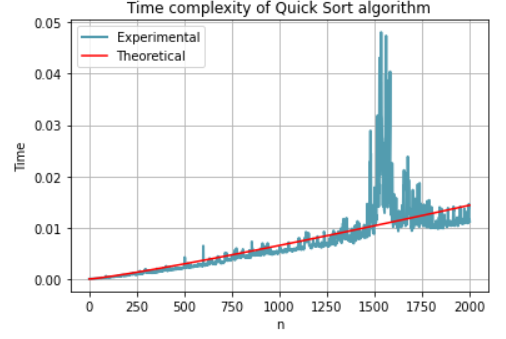


Fig. 7 – Time complexity of Quick Sort algorithm

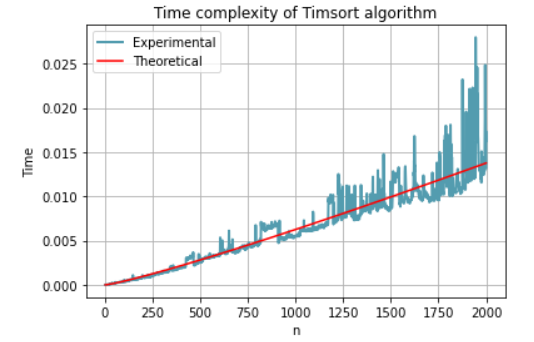


Fig.6 – Time complexity of Timsort algorithm

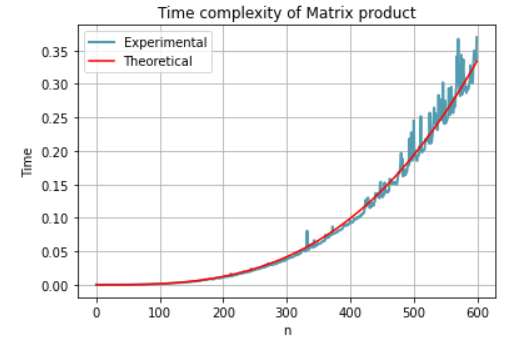


Fig. 9 – Time complexity of product of matrices algorithm

**Conclusion**

**Appendix**