Task 1. Experimental time complexity analysis

Aim

Experimental study of the time complexity for different algorithms

Problems

I. Generate an n-dimensional random vector $\boldsymbol{v} = [v_1, v_2, \dots, v_n]$ with non-negative elements and determine the running time of a procedure or a function f (i.e., an algorithm) realizing certain mathematical calculations over the vector. Use the following procedures and functions:

1) $f(\boldsymbol{v}) = const$ (constant function)
2) $f(\boldsymbol{v}) = \sum_{k=1}^{n} v_k$ (the sum of elements)
3) $f(\boldsymbol{v}) = \prod_{k=1}^{n} v_k$ (the product of elements)
4) $f(\boldsymbol{v}) = \sqrt{\sum_{k=1}^{n} v_k^2}$ (the Euclidean norm of the elements)
5) Let the elements of $\boldsymbol{v}$ be the coefficients of a polynomial P of degree $n-1$. Calculate the value $P(1.5)$ by a direct calculation of $P(x) = \sum_{k=1}^{n} v_k x^{k-1}$ for $x = 1.5$ (i.e. evaluating each term one by one) and by Horner's method representation of the polynomial:
   $P(x) = v_1 + x\big(v_2 + x(v_3 + \cdots)\big)$ for $x = 1.5$.
6) The bubble sort of the elements of $\boldsymbol{v}$

II. Generate random matrices A and B of size $n \times n$ with non-negative elements and determine the running time of the usual matrix product for A and B.

Requirements

Produce 5 runs of each experiment for 10 uniformly distributed values of n from 1 to 500. Find the average running time for each value of n. Running time can be obtained by using timestamps. Draw a graph showing the dependence of n and the corresponding average running time. Compare the empirical time complexity $T(n)$ with available known time complexity for each procedure or function.

Use any programming language you want.

Report should include a pdf document with

- Task number and title, your group name, you name and surname, the date of the report
- Code, graphs and short analysis for each algorithm (including comparison with available known complexity)

Reports should be sent to chunaev@itmo.ru two weeks at latest after the task is given