

Лабораторная работа №9

Тема работы: сложность алгоритмов.

Задание 1

Построить зависимость между количеством элементов и количеством шагов для алгоритмов со сложностью $O(1)$, $O(\log n)$, $O(n^2)$, $O(2^n)$. Сравнить сложность данных алгоритмов.

Задание 2

Для каждого из следующих фрагментов кода определите временную сложность:

<pre>n = 100 for i in range(n): print(i)</pre>
<pre>n = 100 for i in range(n): for j in range(n): print(i, j)</pre>
<pre>n = 100 x = 5 + 3 print(x)</pre>
<pre>n = 100 sum = 0 for i in range(n): for j in range(i, n): sum += 1 print(sum)</pre>
<pre>n = 100 result = 0 for i in range(n): result += i print(result)</pre>
<pre>def factorial(n): if n == 0: return 1 return n * factorial(n - 1)</pre>
<pre>n = 100 count = 0 i = 1 while i <= n: count += 1 i *= 2 print(count)</pre>

Задание 3

Придумать и реализовать алгоритмы, имеющие сложность $O(1)$, $O(n)$, $O(n^2)$, $O(n^3)$.

Задание 4

Изучить и реализовать алгоритм двоичного (бинарного) поиска.