AHMET FIRAT GAMSIZ – 2020400180
ALİ TARIK ŞAHİN – 2020400207

Table of Contents

AHMET FIRAT GAMSIZ – 2020400180
ALİ TARIK ŞAHİN – 2020400207

THEORETICAL ANALYSIS

Basic operation is the comparison marked as (1)

Analyze B(n)

Since the first statement always run unconditionally in every loop, complexity is same for best, average and worst cases.

$$B(n) = \sum_{i=0}^{n-1} 1 = n \in \theta(n)$$

Analyze W(n)

$$W(n) = \sum_{i=0}^{n-1} 1 = n \in \theta(n)$$

Analyze A(n)

$$A(n) = \sum_{i=0}^{n-1} 1 = n \in \theta(n)$$

Basic operations are the two assignments marked as (2)

Analyze B(n)

When array only has 2's in it, none of the (2)'s are executed. Note that using theta notation is incorrect here, since c can't be zero.

$$B(n) = 0 \in O(1)$$

Analyze W(n)

In the worst-case array doesn't contain any 2's (because 0 and 1 run the same code which has more executions).

$$W(n) = \sum_{i=0}^{n-1} n - i = n^2 - \frac{(n-1)\cdot n}{2} = \frac{n\cdot(n+1)}{2} \in \theta(n^2)$$

Analyze A(n)

First (2) runs $(n - i)$ times when arr $[i] = 0$ and this has $1/3$ possibility.
Second (2) runs $(n - i)$ times when arr $[i] = 1$ and this has $1/3$ possibility.
There is $1/3$ possibility for 0 runs.

$$p(I) = \begin{matrix} 1/3 & arr[i] = 0 \\ 1/3 & arr[i] = 1 \\ 1/3 & arr[i] = 2 \end{matrix} \qquad \tau(I) = \begin{matrix} n-i & arc[i] = 0 \\ n-i & arr[i] = 1 \\ 0 & arc[i] = 2 \end{matrix}$$

$$A(n) = \sum_{i=0}^{n-1} \sum_{I\in\tau} \tau(I)p(I)$$

$$= \sum_{i=0}^{n-1} \frac{1}{3}\cdot(n-i) + \frac{1}{3}(n-i) + \frac{1}{3}\cdot 0$$

$$= \sum_{i=0}^{n-1} \frac{2}{3}\cdot n - \frac{2}{3}i = \frac{2}{3}n^2 - \frac{2}{3}\frac{(n-1)n}{2}$$

$$= \frac{2n^2 - n^2 + n}{3} = \frac{n^2+n}{3} = \frac{n(n+1)}{3} \in \theta(n^2)$$

AHMET FIRAT GAMSIZ – 2020400180
ALİ TARIK ŞAHİN – 2020400207

Basic operations are the two comparisons marked as (3)

Analyze B(n)

When array only has 1's in it, none of the (3)'s are executed. Note that using theta notation is incorrect here, since c can't be zero.

$$B(n) = 0 \in O(1)$$

Analyze W(n)

$$W(n) = \max((\lfloor \log_3 n \rfloor + 2) \cdot (n - i), n + 1)$$

- At the first expression plus 2 comes from the fact that both 3^0 being 1 and when it becomes 0 that is checked too.
- At the second expression plus 1 comes from $(n - 1 + 1)$ from 0 to $n - 1$ and $n < n$ is also checked too.

At the beginning, first one is bigger, but for higher values of i second expression becomes bigger as $(n - i)$ approaches constant.

We need to find the point where this max switches. Call the point "s" where:

$$(\lfloor \log_3 n \rfloor + 2)(n - s) = n + 1$$

$$n\lfloor \log_3 n \rfloor + 2n - s(\lfloor \log_3 n \rfloor + 2) = n + 1$$

Let $\lfloor \log_3 n \rfloor = f(n)$

$$s = \frac{n \cdot f(n) + n - 1}{f(n) + 2}$$

s may not be integer so use the floor of it:

$$\sum_{i=0}^{\lfloor s \rfloor} (f(n) + 2)(n - i) + \sum_{i=\lfloor s \rfloor + 1}^{n-1} n + 1$$

$$= \left(\sum_{i=0}^{\lfloor s \rfloor} f(n) \cdot n + 2n - (f(n) + 2)i \right) + (n - \lfloor s \rfloor - 1 \Bigg) (n + 1)$$

$$= (\lfloor s \rfloor + 1)(f(n) \cdot n + 2n) - (f(n) + 2)\frac{\lfloor s \rfloor(\lfloor s \rfloor + 1)}{2} + n^2 - n\lfloor s \rfloor - n + n - \lfloor s \rfloor - 1$$

$$= \lfloor s \rfloor f(n) \cdot n + f(n) \cdot n + \lfloor s \rfloor n + 2n - \frac{1}{2} f(n)\lfloor s \rfloor^2 + \frac{1}{2} f(n)\lfloor s \rfloor + \lfloor s \rfloor^2 - 2\lfloor s \rfloor + n^2 - 1$$

$$= \left\lfloor \frac{n \cdot f(n) + n - 1}{f(n) + 2} \right\rfloor f(n) \cdot n + f(n) \cdot n + \left\lfloor \frac{n \cdot f(n) + n - 1}{f(n) + 2} \right\rfloor n + 2n$$
$$- \frac{1}{2} f(n) \left\lfloor \frac{n \cdot f(n) + n - 1}{f(n) + 2} \right\rfloor^2 + \frac{1}{2} f(n) \left\lfloor \frac{n \cdot f(n) + n - 1}{f(n) + 2} \right\rfloor$$
$$+ \left\lfloor \frac{n \cdot f(n) + n - 1}{f(n) + 2} \right\rfloor^2 - 2 \left\lfloor \frac{n \cdot f(n) + n - 1}{f(n) + 2} \right\rfloor + n^2 - 1$$

This is the exact number of basic operations. For asymptotic notation let's go one step back and use this equation:

$$= \lfloor s \rfloor f(n) \cdot n + f(n) \cdot n + \lfloor s \rfloor n + 2n - \frac{1}{2} f(n) \lfloor s \rfloor^2 + \frac{1}{2} f(n) \lfloor s \rfloor + \lfloor s \rfloor^2 - 2\lfloor s \rfloor + n^2 - 1$$

Fastest growing term of $\lfloor s \rfloor$ can be taken as n, by ignoring lower order terms and simplify. Also, we can replace $f(n)$ with $\log n$:

$$= \log n \cdot n^2 + \log n \cdot n + n^2 + 2n - \frac{1}{2} \log n \cdot n^2 + \frac{1}{2} \log n \cdot n + n^2 - 2n + n^2 - 1$$

$$= \frac{3}{2} \log n \cdot n^2 + \frac{3}{2} \log n \cdot n + 3n^2 - 1$$

Ignoring lower order terms and constants:

$$W(n) \in \theta(n^2 \log\ n)$$

Analyze A(n)

We can split the algorithm and find expected values individually.

$$A(n) = E[\tau] = E[\tau_1(n)] + E[\tau_2(n)] + E[\tau_3(n)]$$

$$\tau_1(n) = \textstyle\sum_{i=0}^{n-1} \sum_{j=i}^{n-1} (\log_3\ n + 2)$$

$$= \sum_{i=0}^{n-1} (n - i)(\log_3 n + 2) = \sum_{i=0}^{n-1} n \log_3 n + 2n - i(\log_3 n + 2)$$
$$= n^2 \log_3 n + 2n^2 - \frac{n \cdot (n-1)}{2} (\log_3 n + 2)$$
$$= n^2 \log_3 n + 2n^2 - \frac{n^2 \log_3 n}{2} - n^2 - \frac{n \log_3 n}{2} - n$$
$$= \frac{n^2 \log_3 n}{2} + n^2 - \frac{n \log_3 n}{2} - n$$
$$\tau_2(n) = 0$$
$$\tau_3(n) = \sum_{i=0}^{n-1} \sum_{p=0}^{n} 1 = n \cdot (n+1) = n^2 + n$$
$$A(n) = E[\tau] = \frac{1}{3}\tau_1(n) + \frac{1}{3}\tau_2(n) + \frac{1}{3}\tau_3(n) = \frac{n^2 \log_3 n}{6} + \frac{n^2}{3} - \frac{n \log_3 n}{6} - \frac{n}{3} + \frac{0}{3} + \frac{n^2}{3} + \frac{n}{3}$$

Ignoring lower order terms and constants:

AHMET FIRAT GAMSIZ – 2020400180
ALİ TARIK ŞAHİN – 2020400207

$$A(n) \in \theta(n^2 \log n)$$

Basic operations are the three assignments marked as (4)

Analyze B(n)

B(n) is found by assuming all the values in the array as zero because if we analyze the blocks inside the outer most loop, we can see that first block grows slower than the other two even when it works as the slowest of its overall, which is when i = 0. In this case, it roughly makes nlogn operations, which is less than n^2 of the third block and even less than the $n^3 logn$ of the second block. However, there is one more case to be considered, which is when the value of i is close to n. In this case, while the third block still operates on n^2 operations, the second block reduces to nlogn operation. Luckily, the first block also works better, and operation count reduces to logn, which is still the least operation count.

B(n) = $\sum_{i=0}^{n-1} \sum_{j=i}^{n-1} \lfloor \log_3 n \rfloor + 1$ (operation count of the first block)

$= \sum_{i=0}^{n-1} (n-i)(\lfloor \log_3 n \rfloor + 1)$

$= \frac{n^2 \lfloor \log_3 n \rfloor}{2} + \frac{n^2}{2} + \frac{n \lfloor \log_3 n \rfloor}{2} + \frac{n}{2}$

B(n) $\in \theta(n^2 logn)$

Analyze W(n)

W(n) = Max$\{first\ block,\ middle\ block,\ last\ block\}$

W(n) = Max$\left\{\sum_{j=i}^{n-1} \lfloor \log_3 n \rfloor + 1, \sum_{m=i}^{n-1} \sum_{l=m}^{n-1} \sum_{t=1}^{n} \sum_{z=1}^{\lceil n/t \rceil} 1, \sum_{p=0}^{n-1} \sum_{j=0}^{p^2-1} 1\right\}$

First expression: $\sum_{j=i}^{n-1} \lfloor \log_3 n \rfloor + 1 = (n-i)(\lfloor \log_3 n \rfloor + 1)$

Second expression: $\sum_{m=i}^{n-1} \sum_{l=m}^{n-1} \sum_{t=1}^{n} \sum_{z=1}^{\lceil n/t \rceil} 1 = \sum_{m=i}^{n-1} \sum_{l=m}^{n-1} \sum_{t=1}^{n} \lceil n/t \rceil$

assuming $\lceil n/t \rceil \sim \frac{n}{t}$ for simplicity of the upcoming operations (this approximation is necessary to continue the calculations), also we showed the harmonic series in the lecture.

$= \sum_{m=i}^{n-1} \sum_{l=m}^{n-1} n \cdot \ln n$

$= \sum_{m=i}^{n-1} (n-m) n \ln n$

$= \sum_{m=i}^{n-1} n^2 \cdot \ln n - mn \cdot \ln n$

$= (n-i) n^2 \ln n - \frac{(n-i)(n-1+i)}{2} n \ln n$

$= n^3 \ln n - in^2 \ln n - \frac{n^3 \ln n}{2} + \frac{in^2 \ln n}{2} + \frac{n^2 \ln n}{2} - \frac{in \ln n}{2} - \frac{in^2 \ln n}{2} + \frac{i^2 n \ln n}{2}$

$= \frac{n^3 lnn}{2} - in^2 lnn + \frac{n^2 lnn}{2} + \frac{i^2 n lnn}{2} - \frac{in lnn}{2}$

Third expression: $\sum_{p=0}^{n-1} \sum_{j=0}^{p^2-1} 1 = \sum_{p=0}^{n-1} p^2 = \frac{(n-1) \cdot n \cdot (2n-1)}{6} = \frac{2n^3 - 3n^2 + n}{6}$

W(n) = Max$\left\{(n-i)(\lfloor \log_3 n \rfloor + 1), \frac{n^3 lnn}{2} - in^2 lnn + \frac{n^2 lnn}{2} + \frac{i^2 n lnn}{2} - \frac{in lnn}{2}, \frac{2n^3 - 3n^2 + n}{6}\right\}$,

AHMET FIRAT GAMSIZ – 2020400180
ALİ TARIK ŞAHİN – 2020400207

First block clearly grows slower than the other two ($nlogn < n^3logn \ and \ nlogn < n^3$). Hence, the middle block and the last block will race. There will be a break point in a specific number of i, now find this number that equates the operation count middle and last blocks.

$$\frac{n^3 lnn}{2} - in^2 lnn + \frac{n^2 lnn}{2} + \frac{i^2 nlnn}{2} - \frac{inlnn}{2} = \frac{2n^3-3n^2+n}{6}$$

Divide to make i^2's coefficient 1 and rearrange terms into perfect square

$$i^2 - (2n+1)i = -n^2 - n + \frac{2n^2}{3lnn} - \frac{n}{lnn} + \frac{1}{3lnn}$$

$$(i-(n+1/2))^2 = \frac{2n^2}{3lnn} - \frac{n}{lnn} + \frac{1}{3lnn} + 1/4$$

$$|i - n - 1/2| = \sqrt{\frac{2n^2-3n+1}{3lnn} + 1/4}$$

* n is always greater than i, hence multiply absolute value with -1

$$i = n + \frac{1}{2} - \sqrt{\frac{2n^2-3n+1}{3lnn} + 1/4}$$

Since i might not be integer, we need to take the floor.
Let s = $\lfloor i \rfloor$

$$W(n) = \sum_{i=0}^{s} \frac{n^3 lnn}{2} - in^2 lnn + \frac{n^2 lnn}{2} + \frac{i^2 nlnn}{2} - \frac{inlnn}{2} + \sum_{i=s+1}^{n-1} \frac{2n^3-3n^2+n}{6}$$

$$W(n) = (s+1)\left(\frac{n^3 lnn}{2} + \frac{n^2 lnn}{2}\right) - \frac{s(s+1)}{2}\left(n^2 lnn + \frac{nlnn}{2}\right) + \frac{s(s+1)(2s+1)}{6}\left(\frac{nlnn}{2}\right) + (n-1-s)\left(\frac{2n^3-3n^2+n}{6}\right)$$

Note that this is the exact count.
Take the fastest growing term in s to find complexity class.

$$W(n) \sim (n+1)\left(\frac{n^3 lnn}{2} + \frac{n^2 lnn}{2}\right) - \frac{n(n+1)}{2}\left(n^2 lnn + \frac{nlnn}{2}\right) + \frac{n(n+1)(2n+1)}{6}\left(\frac{nlnn}{2}\right) + (n-1-n)\left(\frac{2n^3-3n^2+n}{6}\right)$$

Again, take the fastest growing terms, ignoring little terms.

$$W(n) \sim \left(\frac{n^4 lnn}{2} + \frac{n^3 lnn}{2}\right) - \left(\frac{n^4 lnn}{2} + \frac{n^3 lnn}{2}\right) + \left(\frac{2n^4 lnn}{2}\right) + \left(\frac{2n^3-3n^2+n}{6}\right)$$

$$W(n) \sim (n^4 lnn) + \left(\frac{2n^3-3n^2+n}{6}\right)$$

$$W(n) \in \theta(n^4 logn)$$

Analyze A(n)

All the blocks have the same probability of execution. Since we have 3 blocks, each has 1/3 probability to be executed.

A(n)=

+ 1/3 * $\sum_{i=0}^{n-1}(n-i)(\lfloor \log_3 n \rfloor + 1)$ (first block)

+ 1/3 * $\sum_{i=0}^{n-1} \frac{n^3 lnn}{2} - in^2 lnn + \frac{n^2 lnn}{2} + \frac{i^2 nlnn}{2} - \frac{inlnn}{2}$ (second block)

+ 1/3 * $\sum_{i=0}^{n-1} \frac{2n^3-3n^2+n}{6}$ (third block)

$\tau_1(n) = \sum_{i=0}^{n-1}(n-i)(\lfloor \log_3 n \rfloor + 1)$ = $\frac{n^2\lfloor \log_3 n \rfloor}{2} + \frac{n^2}{2} + \frac{n\lfloor \log_3 n \rfloor}{2} + \frac{n}{2}$

$\tau_2(n) = \sum_{i=0}^{n-1} \frac{n^3 lnn}{2} - in^2 lnn + \frac{n^2 lnn}{2} + \frac{i^2 nlnn}{2} - \frac{inlnn}{2}$ = $(n)\left(\frac{n^3 lnn}{2} + \frac{n^2 lnn}{2}\right) - \frac{(n-1)\,(n)}{2}\left(n^2 lnn + \frac{nlnn}{2}\right) + \frac{(n-1)(n)(2n-1)}{6}\left(\frac{nlnn}{2}\right)$

$\tau_3(n) = \sum_{i=0}^{n-1} \frac{2n^3-3n^2+n}{6}$ = $n\frac{2n^3-3n^2+n}{6}$

Put found values in A(n)

A(n) =
+ 1/3 * [$\frac{n^2\lfloor \log_3 n \rfloor}{2} + \frac{n^2}{2} + \frac{n\lfloor \log_3 n \rfloor}{2} + \frac{n}{2}$]
+ 1/3 * [$(n)\left(\frac{n^3 lnn}{2} + \frac{n^2 lnn}{2}\right) - \frac{(n-1)(n)}{2}\left(n^2 lnn + \frac{nlnn}{2}\right) + \frac{(n-1)(n)(2n-1)}{6}\left(\frac{nlnn}{2}\right)$]
+ 1/3 * [$n\frac{2n^3-3n^2+n}{6}$]

Take the fastest growing terms to find complexity class

A(n)~$\frac{n^2\lfloor \log_3 n \rfloor}{6} + \frac{n^2}{6} + \frac{n\lfloor \log_3 n \rfloor}{6} + \frac{n}{6} + \frac{n^4 lnn}{6} + \frac{n^3 lnn}{6} - \frac{n^4 lnn}{6} - \frac{n^3 lnn}{12} + \frac{n^4 lnn}{18}$

A(n)~$\frac{n^2\lfloor \log_3 n \rfloor}{6} + \frac{n^2}{6} + \frac{n\lfloor \log_3 n \rfloor}{6} + \frac{n}{6} + \frac{n^3 lnn}{6} - \frac{n^3 lnn}{12} + \frac{n^4 lnn}{18}$

A(n) ∈ θ($n^4 logn$)

IDENTIFICATION OF BASIC OPERATION(S)

Here, state clearly which operation(s) in the algorithm must be the basic operation(s). Also, you should provide a simple explanation about why you have decided on the basic operation you choose. (1-3 sentences)

It should be the incrementation of y (y = y + 1), which is showed as (4) in the algorithm. Because this operation is the most executed operation and, more importantly, it represents the characteristics of the algorithm. This can also be seen in the graphs of the theoretical analysis.

REAL EXECUTION

Best Case

N Size	Time Elapsed
1	9.5367e-07
5	5.2452e-06

10	1.2159e-05
20	4.5061e-05
30	1.3184e-04
40	2.1791e-04
50	3.3998e-04
60	4.6896e-04
70	6.3514e-04
80	8.2612e-04
90	1.3751e-03
100	1.7960e-03
110	2.3980e-03
120	2.2459e-03
130	2.7289e-03
140	3.0667e-03
150	3.7801e-03

Worst Case

N Size	**Time Elapsed**
1	2.8133e-06
5	1.9574e-05
10	2.1831e-04
20	3.3677e-03
30	1.7939e-02
40	4.8414e-02
50	1.3249e-01
60	2.9760e-01
70	5.3822e-01
80	8.5448e-01
90	1.4875e+00
100	2.1897e+00
110	3.4051e+00
120	4.7076e+00
130	6.4297e+00
140	8.8179e+00
150	1.1790e+01

Average Case

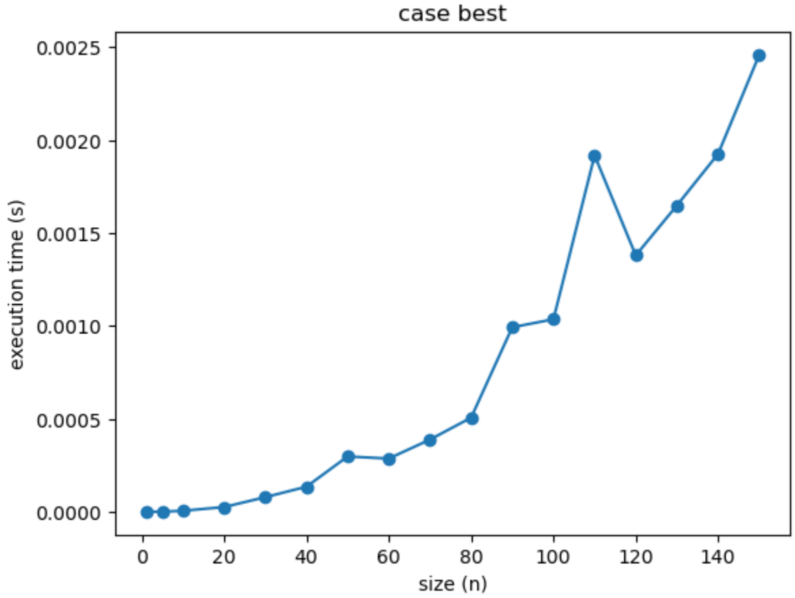
N Size	**Time Elapsed**
1	3.0994e-06
5	5.0306e-05
10	5.7864e-04
20	*1.3564e-02*
30	5.4389e-02
40	*1.4546e-01*
50	3.3477e-01
60	6.9422e-01
70	1.3073e+00

80	2.2459e+00
90	3.6204e+00
100	5.5011e+00
110	8.1777e+00
120	1.1610e+01
130	1.6020e+01
140	2.1531e+01
150	2.8684e+01

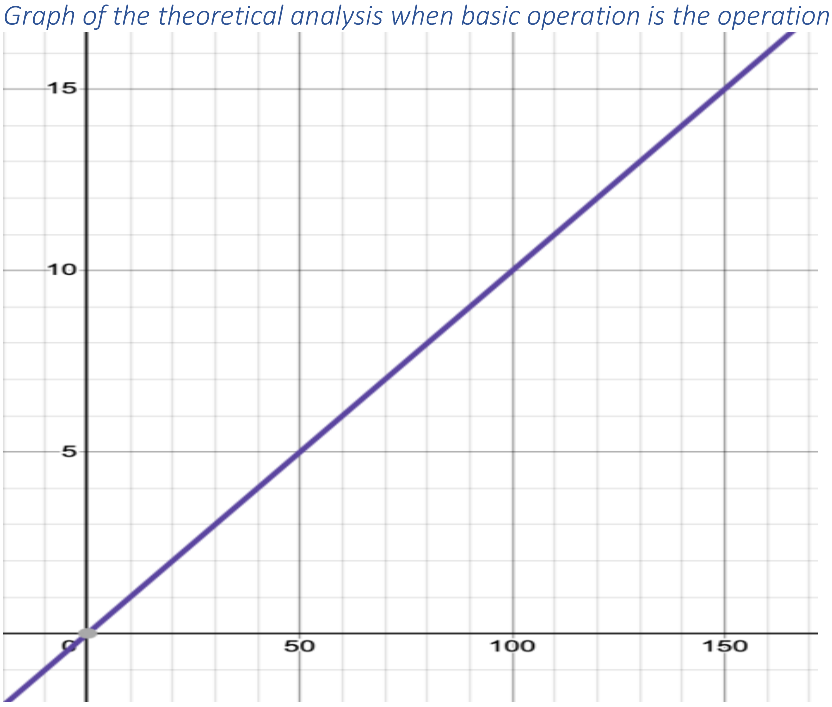
COMPARISON

Best Case

Graph of the real execution time of the algorithm



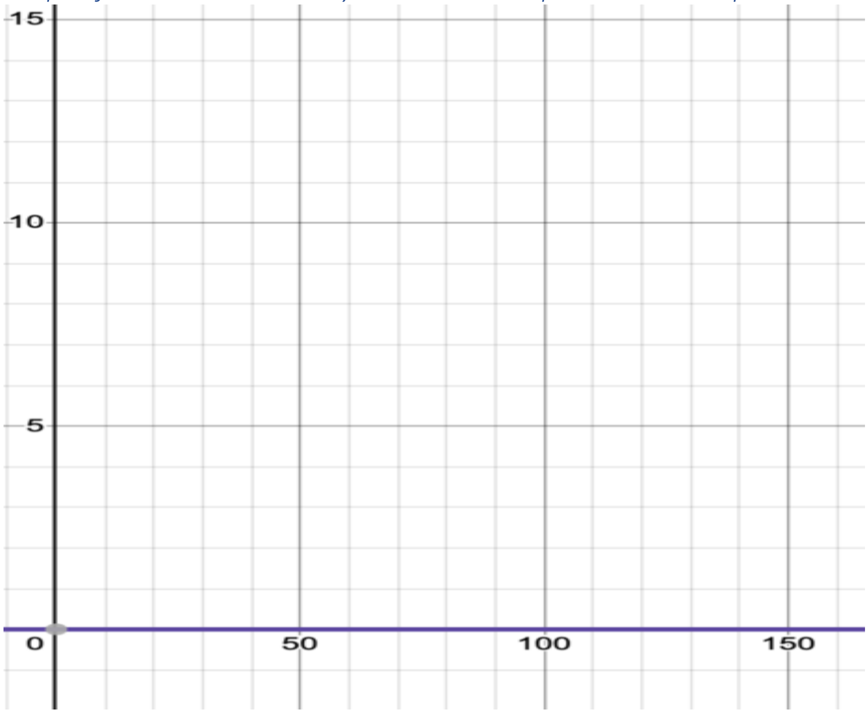
Graph of the theoretical analysis when basic operation is the operation marked as (1)



AHMET FIRAT GAMSIZ – 2020400180
ALİ TARIK ŞAHİN – 2020400207

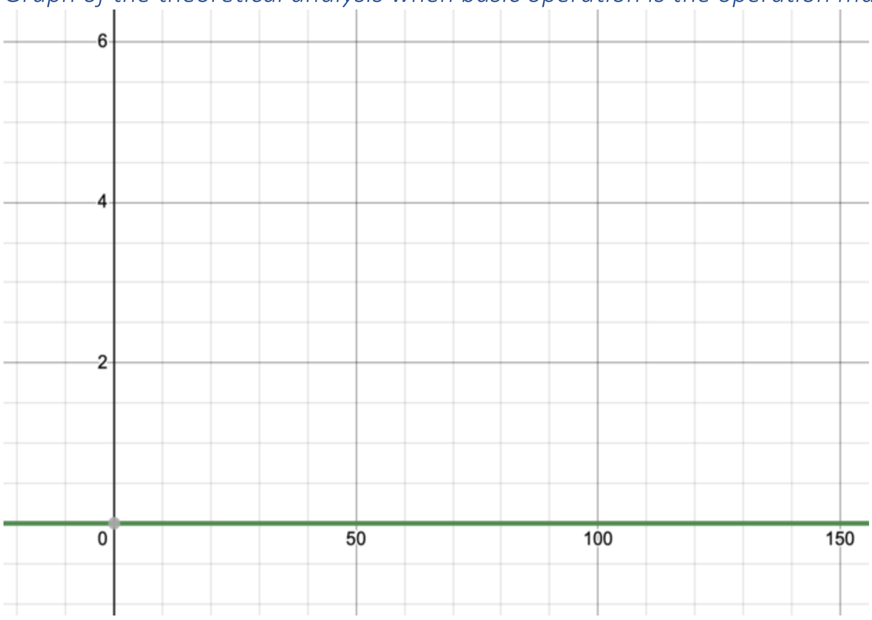
(y axis is complexity, x axis is input size)

Graph of the theoretical analysis when basic operation is the operation marked as (2)



(y axis is complexity, x axis is input size)

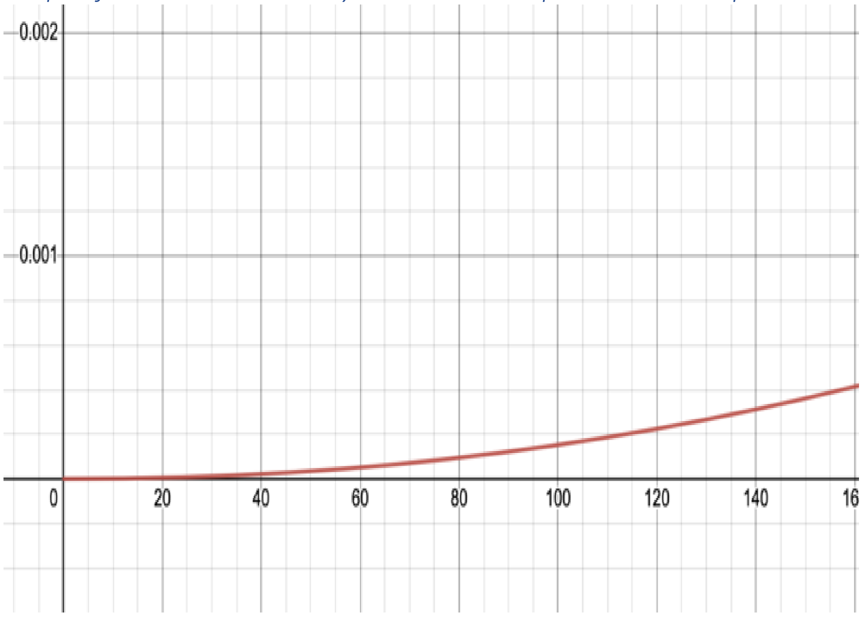
Graph of the theoretical analysis when basic operation is the operation marked as (3)



(y axis is complexity, x axis is input size)

AHMET FIRAT GAMSIZ – 2020400180
ALİ TARIK ŞAHİN – 2020400207

Graph of the theoretical analysis when basic operation is the operation marked as (4)



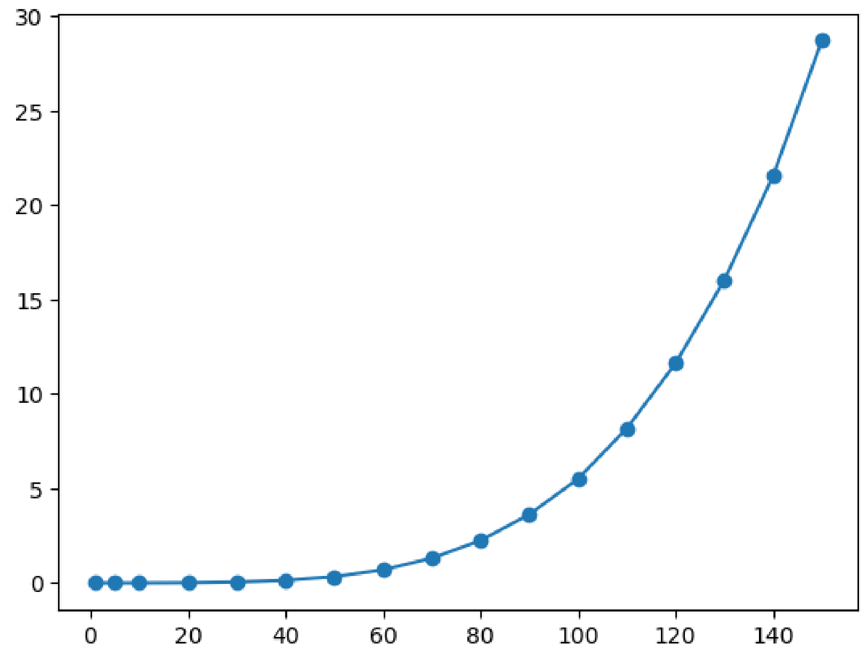
(y axis is complexity, x axis is input size)

Comments

For the first graph, that is the function of best case of the first case. The real output is multiplied with 10 in this case. Linear growth can be observed in the graphical analysis. This basic operation choice does not fit to the real behavior of the algorithm, so we can conclude that it is a bad choice. Similarly, second and the third graphs are zero functions and they also do not reflect the real behavior. In the last graph, values on the y axis need to be multiplied with 10^8. This graph is the most similar graph to real time execution.

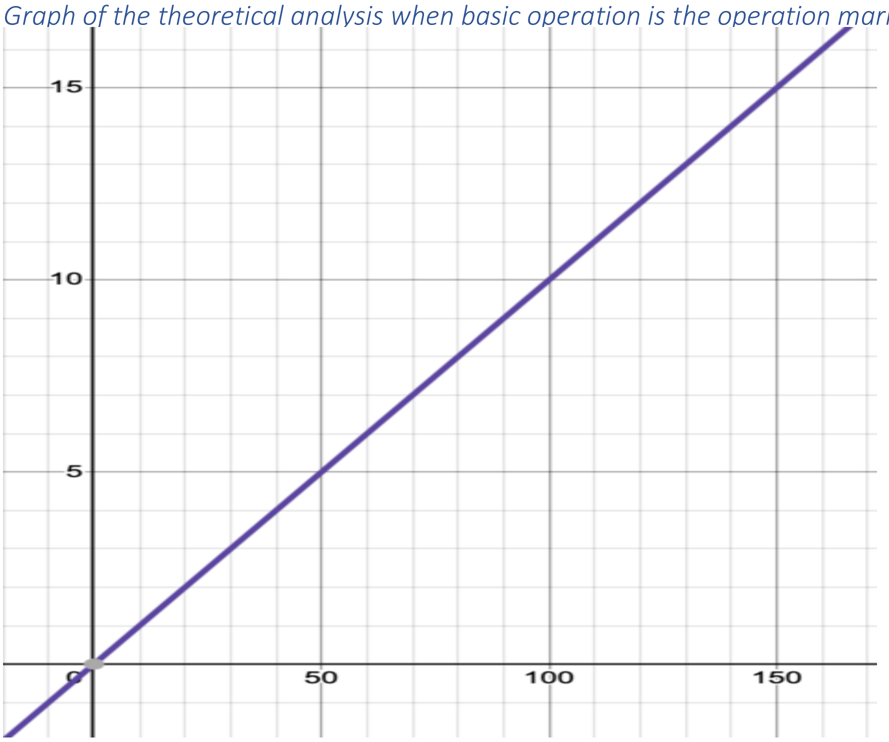
Worst Case

Graph of the real execution time of the algorithm



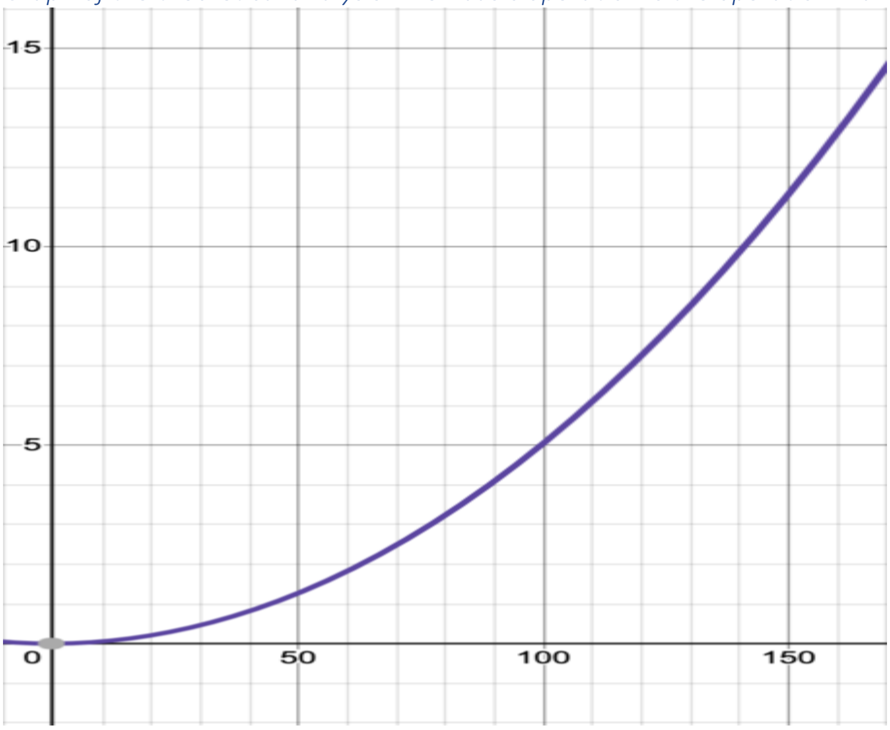
AHMET FIRAT GAMSIZ – 2020400180
ALİ TARIK ŞAHİN – 2020400207

Graph of the theoretical analysis when basic operation is the operation marked as (1)



(y axis is complexity, x axis is input size)

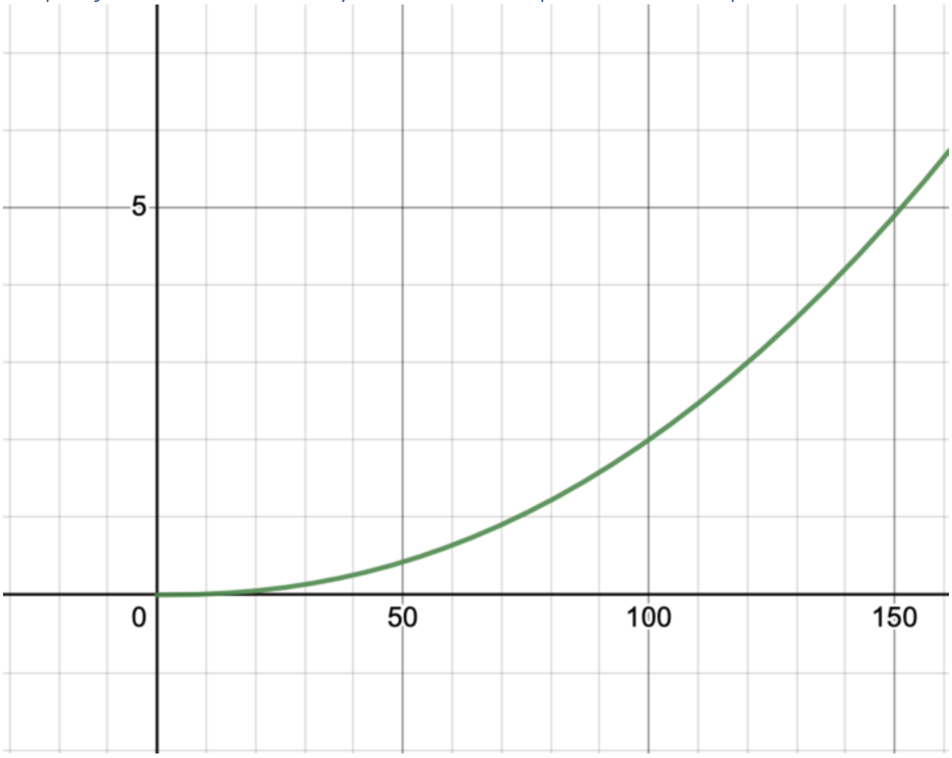
Graph of the theoretical analysis when basic operation is the operation marked as (2)



(y axis is complexity, x axis is input size)

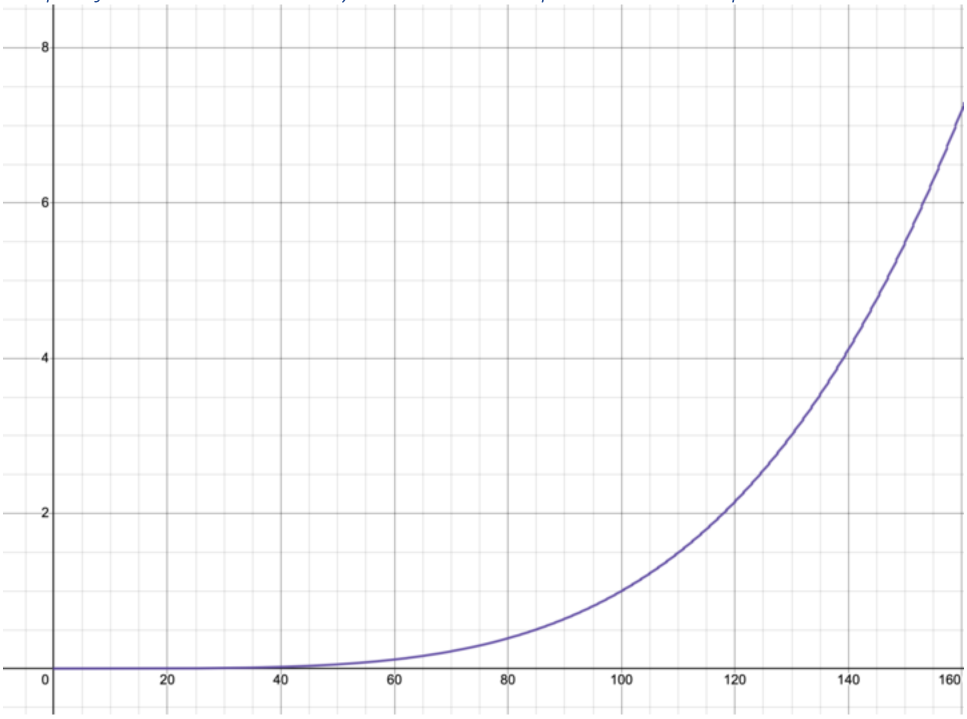
AHMET FIRAT GAMSIZ – 2020400180
ALİ TARIK ŞAHİN – 2020400207

Graph of the theoretical analysis when basic operation is the operation marked as (3)



(y axis is complexity, x axis is input size)

Graph of the theoretical analysis when basic operation is the operation marked as (4)



(y axis is complexity, x axis is input size)

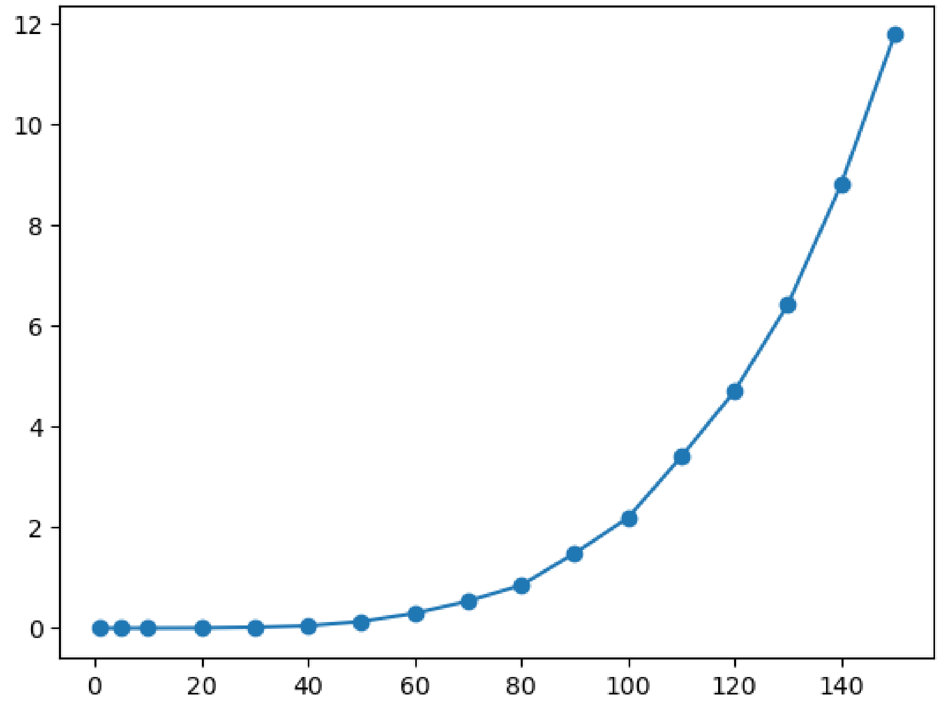
Comments

First case is again linear, which is not even close to the real behavior; however, the other two might be good choices. Assume that values at the y axis will be multiplied by 10^3 in (2) 10^4

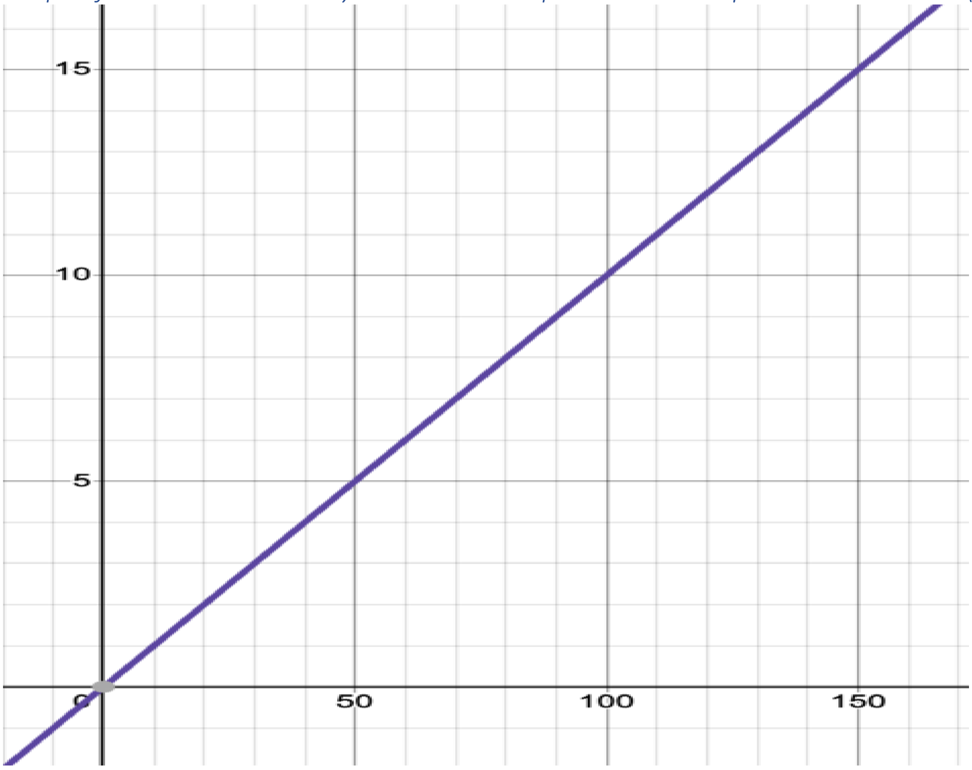
in (3) 10^8 in (4). Moreover, it can be observed that the last graph almost has the exact match with the real time execution. Hence, again basic operation marked as (4) is the best choice.

Average Case

Graph of the real execution time of the algorithm



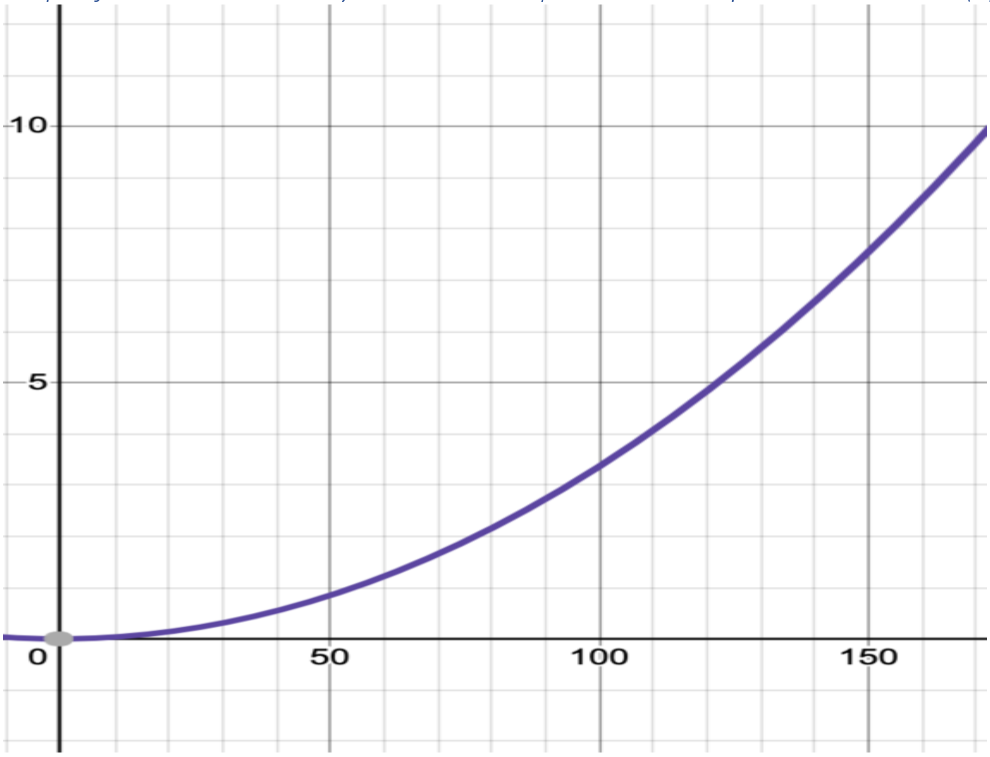
Graph of the theoretical analysis when basic operation is the operation marked as (1)



(y axis is complexity, x axis is input size)

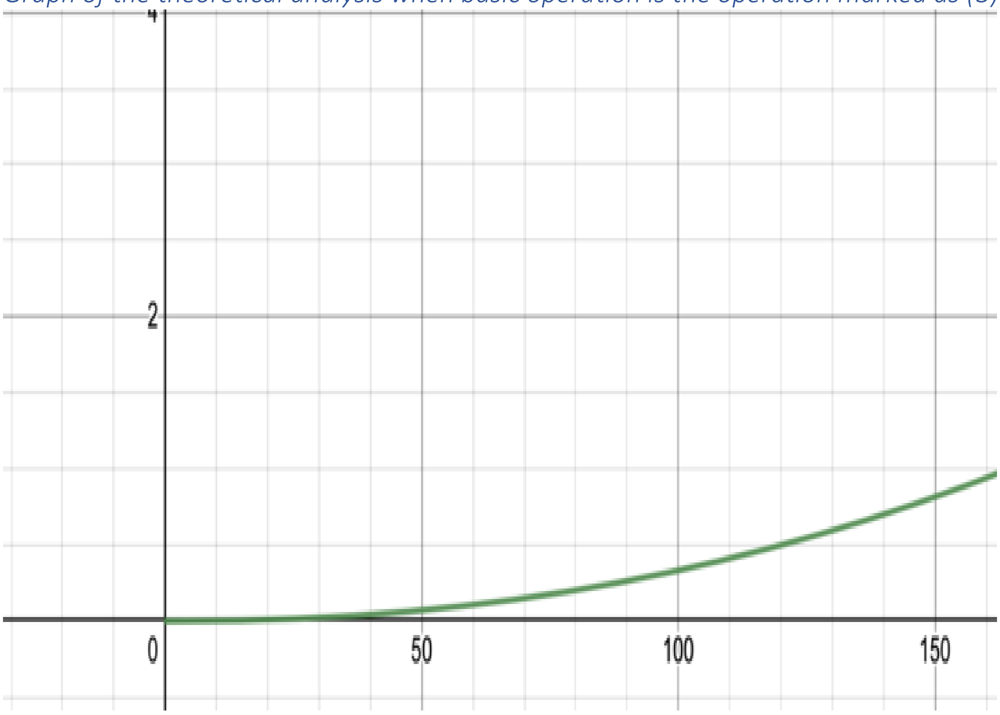
AHMET FIRAT GAMSIZ – 2020400180
ALİ TARIK ŞAHİN – 2020400207

Graph of the theoretical analysis when basic operation is the operation marked as (2)



(y axis is complexity, x axis is input size)

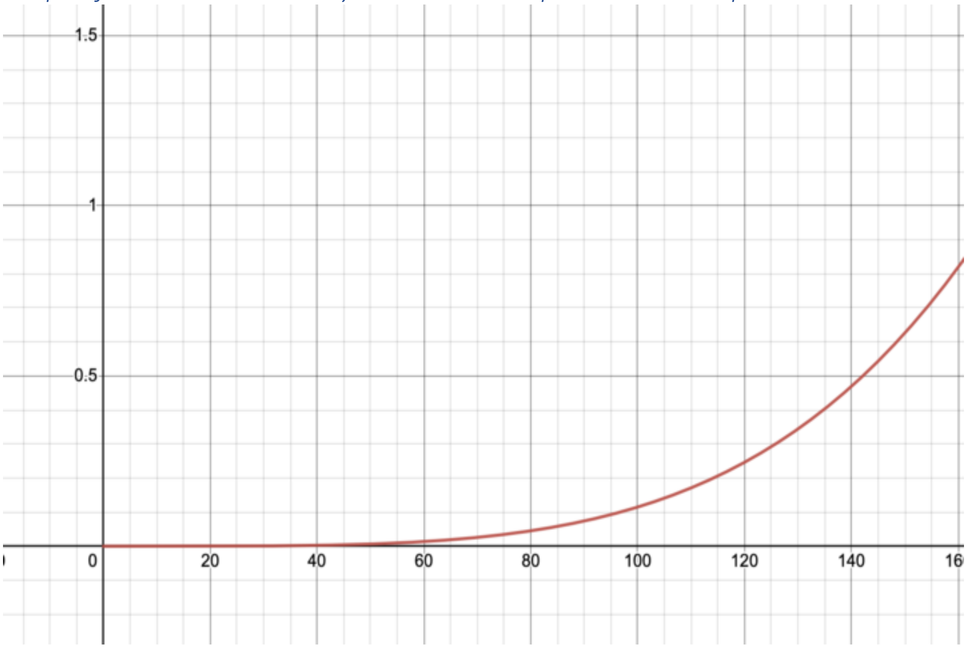
Graph of the theoretical analysis when basic operation is the operation marked as (3)



(y axis is complexity, x axis is input size)

AHMET FIRAT GAMSIZ – 2020400180
ALİ TARIK ŞAHİN – 2020400207

Graph of the theoretical analysis when basic operation is the operation marked as (4)



(y axis is complexity, x axis is input size)

Comments

First case is again linear, which is not we are looking for. Other three cases are similar; however, when we take a closer look at the real behavior of the function, we see that it goes on like a constant function until 50-60 inputs and then it grows in a fast manner. This behavior is also explicit in the last function, which again shows that basic operation should be chosen as (4).