

به نام خدا

## تمرین 1

محمد مهدی صفرمحمدلو 95440296

طراحی الگوریتم استاد بختیاری

- روش A1 : محاسبه با روش مستقیم (محاسبه فاکتوریل با حلقه)
- روش A2 : محاسبه با روش مستقیم (محاسبه فاکتوریل با بازگشتی)
- روش A3 : محاسبه با روش مستقیم (محاسبه فاکتوریل با بازگشتی و تابع یادآور)
- روش A4 : محاسبه با روش مستقیم (محاسبه فاکتوریل با برنامه نویسی پویا)
- روش B : محاسبه با روش بازگشتی
- روش C : محاسبه با روش بازگشتی و تابع یادآور
- روش D : محاسبه با روش برنامه نویسی پویا

سیستم تست شده :

Corei5-6200U

8G Ram

زبان برنامه نویسی استفاده شده : Kotlin

بر روی 1.7 JVM (java virtual machine)

هر تست 100 بار تکرار شده و میانگین آن به نمایش درآمده  
تست به این گونه انجام شده است که مقدار  $N$  در فرمول ،  $NK$  است  
و مقدار  $K$  در فرمول ،  $NK/2$  است

یعنی همیشه  $K$  نصف  $N$  میباشد تا کشیدن نمودار و مقایسه راحت تر شود و اعداد به میلی ثانیه می باشد

\*روش B (بازگشتی) در  $n = 40$  و  $k = 20$  رام نشدنی میشود

\*روش C (بازگشتی و تابع یادآور) و روش D (برنامه نویسی پویا) در  $n = 5120$  و  $k = 2560$  رام نشدنی میشوند

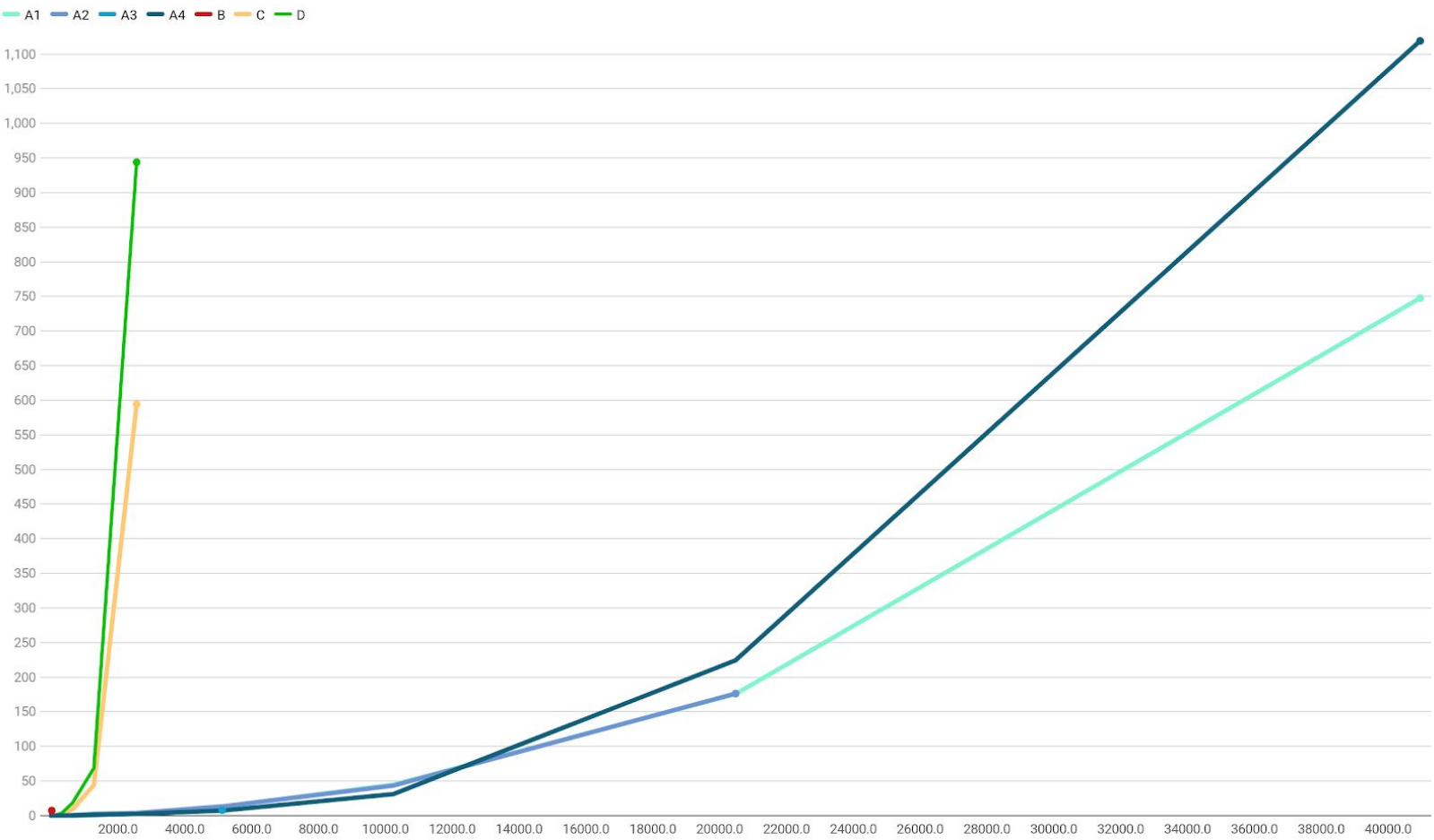
\*روش A3 (روش مستقیم (محاسبه فاکتوریل با بازگشتی و تابع یادآور)) در  $n = 10240$  و  $k = 5120$  با خطای Stack Overflow روبرو شد

\*روش A2 (روش مستقیم (محاسبه فاکتوریل با بازگشتی)) در  $n = 10240$  و  $k = 5120$  با خطای Stack Overflow روبرو شد

1	NK	A1	A2	A3	A4	B	C	D
2	5	0.75	0.01	0.040000	0.01	0.01	0.00	0.00
3	10	0.02	0.01	0.000000	0.01	0.03	0.01	0.06
4	20	0.04	0.02	0.020001	0.01	7.38	0.03	0.05
5	40	0.04	0.01	0.019999	0.02	null	0.12	0.31
6	80	0.09	0.03	0.019999	0.06	null	0.56	0.45
7	160	0.06	0.05	0.050002	0.03	null	0.59	1.01
8	320	0.07	0.07	0.100001	0.23	null	2.19	3.69
9	640	0.48	0.38	0.190001	0.12	null	9.28	18.34
10	1,280	2.88	2.14	0.930002	1.35	null	44.18	68.69
11	2,560	3.48	3.88	2.579999	2.51	null	594.56	943.99
12	5,120	12.84	13.26	7.750001	7.28	null	null	null
13	10,240	44.66	43.35	null	31.20	null	null	null
14	20,480	175.73	176.29	null	224.33	null	null	null
15	40,960	747.55	null	null	1,119.12	null	null	null

## جدول مقایسه روش ها

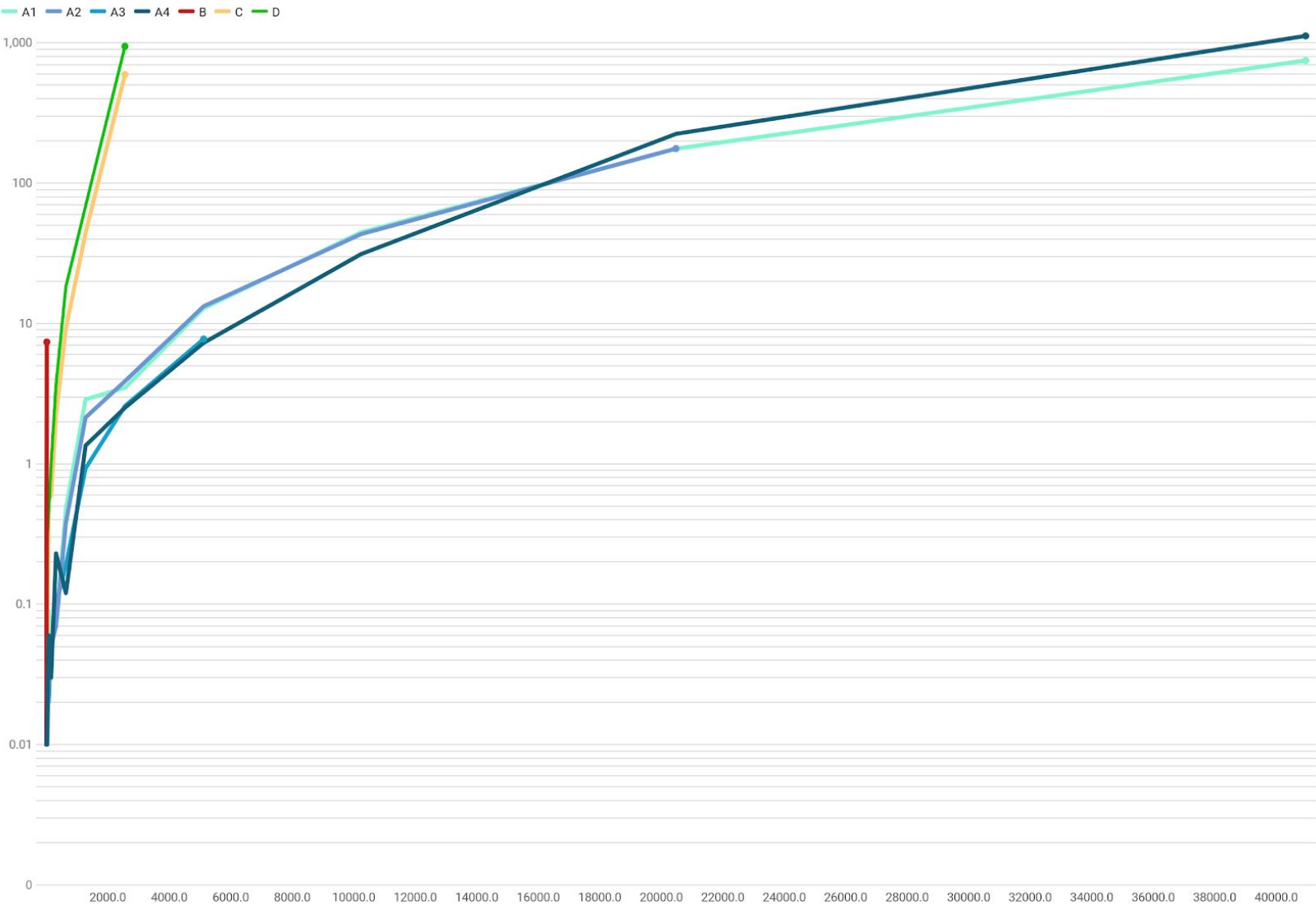
### ZaribDobandi Algorithms



جدول مقایسه روش ها با استفاده از لگاریتم برای مقایسه بهتر

ZaribDobandi Algorithms

Logarithmic chart



در همه ی روش ها برای محاسبه ی اعداد بزرگتر به دلیل محدودیت های Int و Long از کلاس BigInteger جاوا استفاده شد تا بتواند اعداد بسیار بزرگ را محاسبه کند

بزرگترین عددی که محاسبه شد ، با  $n$  هفت رقمی صورت پذیرفت

$$N = 1000000$$

$$K = 500000$$

محاسبه ی این ضریب دوبرندی 727239 میلی ثانیه معادل با 12.12 دقیقه صورت پذیرفت و عدد بسیار بزرگی که **301 هزار رقمی** است را با روش A1 که سریع ترین روش بود تولید کرد  
فایل این عدد :

<https://raw.githubusercontent.com/mahdisml/TarahiAlgorithmTamrin1/master/BiggestNumber.txt>

سورس برنامه :

<https://github.com/mahdisml/TarahiAlgorithmTamrin1/blob/master/Main.kt>

اجرای زنده کد :

<https://pl.kotl.in/MhxxgNlKuS>

محمد مهدی صفرمحمدلو