

MODULE 3

(EASY)

Imas have a list of products:

Product Awal			
Nama Produk		Harga	Rating

- Ultramilk	Rp	7.000	4,8
- Clevo	Rp	3.000	2,5
- Bear Brend	Rp	10.000	3,5
- Greenfield	Rp	7.500	4,0
- Oreo	Rp	12.000	4,8
- Indomilk	Rp	8.000	5,0
- Japota	Rp	10.000	4,3
- Chocolatos Keju	Rp	7.000	2,5
- Sarimi Isi 2	Rp	4.000	4,4
- Pillow	Rp	5.000	5,0
- Lifebuoy	Rp	4.000	4,7
- Dettol	Rp	4.500	4,9
- Chitato	Rp	13.000	5,0
- Supermi	Rp	2.800	4,2
- Giv	Rp	3.000	4,4
- Shinzui	Rp	5.000	4,8
- Rinso	Rp	25.000	4,8
- Daia	Rp	18.000	4,5
- Nuvo	Rp	3.500	4,5
- Attack	Rp	28.000	4,9
- Indomie Goreng	Rp	3.500	5,0
- So Klin	Rp	22.000	4,6
- Mie Sedaap	Rp	3.000	4,8
- Pop Mie	Rp	6.000	4,6
- Boom	Rp	15.000	4,3
- Teh Botol Sosro	Rp	5.000	4,8
- Aqua 600ml	Rp	3.500	4,9
- Pocari Sweat	Rp	7.000	4,7
- Sprite	Rp	6.000	5,0
- Coca-Cola	Rp	6.000	4,5
- Kopi Instan Gayo	Rp	25.000	4,9
- Teh Melati Celup	Rp	15.000	4,7
- Susu UHT Coklat 1L	Rp	18.000	4,8
- Biskuit Kelapa Renyah	Rp	9.500	4,5
- Keripik Singkong Balado	Rp	12.000	4,6
- Sereal Gandum Madu	Rp	32.000	4,7
- Jus Jambu Merah	Rp	8.000	4,4
- Mie Instan Kari Spesial	Rp	3.500	5,0

Then, Imas want to see data with name “Jus Jambu Merah”, she using both of searching methods she knew, which is Linear Search and Binary Search:

Mencari Product dengan nama : "Jus Jambu Merah" dengan metode Linear Search			
Nama Produk		Harga	Rating

- Jus Jambu Merah	Rp	8.000	4,4
Waktu pencarian: 0,000852 detik			
Mencari Product dengan nama : "Jus Jambu Merah" dengan metode Binary Search			
Nama Produk		Harga	Rating

- Jus Jambu Merah	Rp	8.000	4,4
Waktu pencarian: 0,001285 detik			

Note: Search time can be calculated using the code below, or another algorithm.

```
System.out.println("Mencari Product dengan nama : \"Jus Jambu Merah\" dengan metode Binary Search");
    startTime = System.nanoTime();
    list.binarySearch("Jus Jambu Merah");
    endTime = System.nanoTime();
    waktu = (double) (endTime - startTime)/1000000000.0;
    System.out.printf("Waktu pencarian: %8f detik", waktu);
    System.out.println();
    System.out.println();
```

After that, Imas want to sort her list products based on name, she use both of sorting methods she knew, which is Bubble Sort (Ascending) and Selection Sort (Descending):

```
Mengurutkan Product berdasarkan nama secara Ascending dengan metode Bubble Sort
Nama Produk | Harga | Rating
-----|-----|-----
- Aqua 600ml | Rp 3.500 | 4,9
- Attack | Rp 28.000 | 4,9
- Bear Brend | Rp 10.000 | 3,5
- Biskuit Kelapa Renyah | Rp 9.500 | 4,5
- Boom | Rp 15.000 | 4,3
- Chitato | Rp 13.000 | 5,0
- Chocolatos Keju | Rp 7.000 | 2,5
- Clevo | Rp 3.000 | 2,5
- Coca-Cola | Rp 6.000 | 4,5
- Daia | Rp 18.000 | 4,5
- Dettol | Rp 4.500 | 4,9
- Giv | Rp 3.000 | 4,4
- Greenfield | Rp 7.500 | 4,0
- Indomie Goreng | Rp 3.500 | 5,0
- Indomilk | Rp 8.000 | 5,0
- Japota | Rp 10.000 | 4,3
- Jus Jambu Merah | Rp 8.000 | 4,4
- Keripik Singkong Balado | Rp 12.000 | 4,6
- Kopi Instan Gayo | Rp 25.000 | 4,9
- Lifebuoy | Rp 4.000 | 4,7
- Mie Instan Kari Spesial | Rp 3.500 | 5,0
- Mie Sedaap | Rp 3.000 | 4,8
- Nuvo | Rp 3.500 | 4,5
- Oreo | Rp 12.000 | 4,8
- Pillow | Rp 5.000 | 5,0
- Pocari Sweat | Rp 7.000 | 4,7
- Pop Mie | Rp 6.000 | 4,6
- Rinso | Rp 25.000 | 4,8
- Sarimi Isi 2 | Rp 4.000 | 4,4
- Sereal Gandum Madu | Rp 32.000 | 4,7
- Shinzui | Rp 5.000 | 4,8
- So Klin | Rp 22.000 | 4,6
- Sprite | Rp 6.000 | 5,0
- Supermi | Rp 2.800 | 4,2
- Susu UHT Coklat 1L | Rp 18.000 | 4,8
- Teh Botol Sosro | Rp 5.000 | 4,8
- Teh Melati Celup | Rp 15.000 | 4,7
- Ultramilk | Rp 7.000 | 4,8
Waktu pengurutan: 0,000176 detik
```

Mengurutkan Product berdasarkan nama secara Descending dengan metode Selection Sort			
Nama Produk		Harga	Rating
- Ultramilk	Rp	7.000	4,8
- Teh Melati Celup	Rp	15.000	4,7
- Teh Botol Sosro	Rp	5.000	4,8
- Susu UHT Coklat 1L	Rp	18.000	4,8
- Supermi	Rp	2.800	4,2
- Sprite	Rp	6.000	5,0
- So Klin	Rp	22.000	4,6
- Shinzui	Rp	5.000	4,8
- Sereal Gandum Madu	Rp	32.000	4,7
- Sarimi Isi 2	Rp	4.000	4,4
- Rinso	Rp	25.000	4,8
- Pop Mie	Rp	6.000	4,6
- Pocari Sweat	Rp	7.000	4,7
- Pillow	Rp	5.000	5,0
- Oreo	Rp	12.000	4,8
- Nuvo	Rp	3.500	4,5
- Mie Sedaap	Rp	3.000	4,8
- Mie Instan Kari Spesial	Rp	3.500	5,0
- Lifebuoy	Rp	4.000	4,7
- Kopi Instan Gayo	Rp	25.000	4,9
- Keripik Singkong Balado	Rp	12.000	4,6
- Jus Jambu Merah	Rp	8.000	4,4
- Japota	Rp	10.000	4,3
- Indomilk	Rp	8.000	5,0
- Indomie Goreng	Rp	3.500	5,0
- Greenfield	Rp	7.500	4,0
- Giv	Rp	3.000	4,4
- Dettol	Rp	4.500	4,9
- Daia	Rp	18.000	4,5
- Coca-Cola	Rp	6.000	4,5
- Clevo	Rp	3.000	2,5
- Chocolatos Keju	Rp	7.000	2,5
- Chitato	Rp	13.000	5,0
- Boom	Rp	15.000	4,3
- Biskuit Kelapa Renyah	Rp	9.500	4,5
- Bear Brend	Rp	10.000	3,5
- Attack	Rp	28.000	4,9
- Aqua 600ml	Rp	3.500	4,9

Waktu pengurutan: 0,000107 detik

Based on price:

Mengurutkan Product berdasarkan harga secara Ascending dengan metode Bubble Sort			
Nama Produk		Harga	Rating
- Supermi	Rp	2.800	4,2
- Clevo	Rp	3.000	2,5
- Giv	Rp	3.000	4,4
- Mie Sedaap	Rp	3.000	4,8
- Aqua 600ml	Rp	3.500	4,9
- Indomie Goreng	Rp	3.500	5,0
- Mie Instan Kari Spesial	Rp	3.500	5,0
- Nuvo	Rp	3.500	4,5
- Lifebuoy	Rp	4.000	4,7
- Sarimi Isi 2	Rp	4.000	4,4
- Dettol	Rp	4.500	4,9
- Pillow	Rp	5.000	5,0
- Shinzui	Rp	5.000	4,8
- Teh Botol Sosro	Rp	5.000	4,8
- Coca-Cola	Rp	6.000	4,5
- Pop Mie	Rp	6.000	4,6
- Sprite	Rp	6.000	5,0
- Chocolatos Keju	Rp	7.000	2,5
- Pocari Sweat	Rp	7.000	4,7
- Ultramilk	Rp	7.000	4,8
- Greenfield	Rp	7.500	4,0
- Indomilk	Rp	8.000	5,0
- Jus Jambu Merah	Rp	8.000	4,4
- Biskuit Kelapa Renyah	Rp	9.500	4,5
- Bear Brend	Rp	10.000	3,5
- Japota	Rp	10.000	4,3
- Keripik Singkong Balado	Rp	12.000	4,6
- Oreo	Rp	12.000	4,8
- Chitato	Rp	13.000	5,0
- Boom	Rp	15.000	4,3
- Teh Melati Celup	Rp	15.000	4,7
- Daia	Rp	18.000	4,5
- Susu UHT Coklat 1L	Rp	18.000	4,8
- So Klin	Rp	22.000	4,6
- Kopi Instan Gayo	Rp	25.000	4,9
- Rinso	Rp	25.000	4,8
- Attack	Rp	28.000	4,9
- Sereal Gandum Madu	Rp	32.000	4,7

Waktu pengurutan: 0,000197 detik

Mengurutkan Product berdasarkan harga secara Descending dengan metode Selection Sort			
Nama Produk		Harga	Rating

- Sereal Gandum Madu	Rp	32.000	4,7
- Attack	Rp	28.000	4,9
- Rinso	Rp	25.000	4,8
- Kopi Instan Gayo	Rp	25.000	4,9
- So Klin	Rp	22.000	4,6
- Daia	Rp	18.000	4,5
- Susu UHT Coklat 1L	Rp	18.000	4,8
- Boom	Rp	15.000	4,3
- Teh Melati Celup	Rp	15.000	4,7
- Chitato	Rp	13.000	5,0
- Keripik Singkong Balado	Rp	12.000	4,6
- Oreo	Rp	12.000	4,8
- Bear Brend	Rp	10.000	3,5
- Japota	Rp	10.000	4,3
- Biskuit Kelapa Renyah	Rp	9.500	4,5
- Jus Jambu Merah	Rp	8.000	4,4
- Indomilk	Rp	8.000	5,0
- Greenfield	Rp	7.500	4,0
- Pocari Sweat	Rp	7.000	4,7
- Chocolatos Keju	Rp	7.000	2,5
- Ultramilk	Rp	7.000	4,8
- Coca-Cola	Rp	6.000	4,5
- Pop Mie	Rp	6.000	4,6
- Sprite	Rp	6.000	5,0
- Shinzui	Rp	5.000	4,8
- Teh Botol Sosro	Rp	5.000	4,8
- Pillow	Rp	5.000	5,0
- Dettol	Rp	4.500	4,9
- Lifebuoy	Rp	4.000	4,7
- Sarimi Isi 2	Rp	4.000	4,4
- Aqua 600ml	Rp	3.500	4,9
- Indomie Goreng	Rp	3.500	5,0
- Mie Instan Kari Spesial	Rp	3.500	5,0
- Nuvo	Rp	3.500	4,5
- Mie Sedaap	Rp	3.000	4,8
- Clevo	Rp	3.000	2,5
- Giv	Rp	3.000	4,4
- Supermi	Rp	2.800	4,2
Waktu pengurutan: 0,000043 detik			

Based on rating:

Mengurutkan Product berdasarkan rating secara Ascending dengan metode Bubble Sort			
Nama Produk		Harga	Rating

- Chocolatos Keju	Rp	7.000	2,5
- Clevo	Rp	3.000	2,5
- Bear Brend	Rp	10.000	3,5
- Greenfield	Rp	7.500	4,0
- Supermi	Rp	2.800	4,2
- Boom	Rp	15.000	4,3
- Japota	Rp	10.000	4,3
- Giv	Rp	3.000	4,4
- Jus Jambu Merah	Rp	8.000	4,4
- Sarimi Isi 2	Rp	4.000	4,4
- Biskuit Kelapa Renyah	Rp	9.500	4,5
- Coca-Cola	Rp	6.000	4,5
- Daia	Rp	18.000	4,5
- Nuvo	Rp	3.500	4,5
- Keripik Singkong Balado	Rp	12.000	4,6
- Pop Mie	Rp	6.000	4,6
- So Klin	Rp	22.000	4,6
- Lifebuoy	Rp	4.000	4,7
- Pocari Sweat	Rp	7.000	4,7
- Sereal Gandum Madu	Rp	32.000	4,7
- Teh Melati Celup	Rp	15.000	4,7
- Mie Sedaap	Rp	3.000	4,8
- Oreo	Rp	12.000	4,8
- Rinso	Rp	25.000	4,8
- Shinzui	Rp	5.000	4,8
- Susu UHT Coklat 1L	Rp	18.000	4,8
- Teh Botol Sosro	Rp	5.000	4,8
- Ultramilk	Rp	7.000	4,8
- Aqua 600ml	Rp	3.500	4,9
- Attack	Rp	28.000	4,9
- Dettol	Rp	4.500	4,9
- Kopi Instan Gayo	Rp	25.000	4,9
- Indomilk	Rp	8.000	5,0
- Chitato	Rp	13.000	5,0
- Indomie Goreng	Rp	3.500	5,0
- Mie Instan Kari Spesial	Rp	3.500	5,0
- Pillow	Rp	5.000	5,0
- Sprite	Rp	6.000	5,0
Waktu pengurutan: 0,000162 detik			

Mengurutkan Product berdasarkan rating secara Descending dengan metode Selection Sort

Nama Produk	Harga	Rating
- Sprite	Rp 6.000	5,0
- Pillow	Rp 5.000	5,0
- Mie Instan Kari Spesial	Rp 3.500	5,0
- Indomie Goreng	Rp 3.500	5,0
- Chitato	Rp 13.000	5,0
- Indomilk	Rp 8.000	5,0
- Kopi Instan Gayo	Rp 25.000	4,9
- Dettol	Rp 4.500	4,9
- Attack	Rp 28.000	4,9
- Aqua 600ml	Rp 3.500	4,9
- Rinso	Rp 25.000	4,8
- Oreo	Rp 12.000	4,8
- Mie Sedaap	Rp 3.000	4,8
- Teh Botol Sosro	Rp 5.000	4,8
- Ultramilk	Rp 7.000	4,8
- Susu UHT Coklat 1L	Rp 18.000	4,8
- Shinzui	Rp 5.000	4,8
- Teh Melati Celup	Rp 15.000	4,7
- Lifebuoy	Rp 4.000	4,7
- Pocari Sweat	Rp 7.000	4,7
- Sereal Gandum Madu	Rp 32.000	4,7
- Pop Mie	Rp 6.000	4,6
- So Klin	Rp 22.000	4,6
- Keripik Singkong Balado	Rp 12.000	4,6
- Nuvo	Rp 3.500	4,5
- Daia	Rp 18.000	4,5
- Coca-Cola	Rp 6.000	4,5
- Biskuit Kelapa Renyah	Rp 9.500	4,5
- Giv	Rp 3.000	4,4
- Jus Jambu Merah	Rp 8.000	4,4
- Sarimi Isi 2	Rp 4.000	4,4
- Boom	Rp 15.000	4,3
- Japota	Rp 10.000	4,3
- Supermi	Rp 2.800	4,2
- Greenfield	Rp 7.500	4,0
- Bear Brand	Rp 10.000	3,5
- Chocolatos Keju	Rp 7.000	2,5
- Clevo	Rp 3.000	2,5

Waktu pengurutan: 0,000060 detik

Imas want to see clearly the differents between each sorting algorithm:

```
=====
BUBBLE ASC SORT
=====
Mengurutkan Product berdasarkan harga secara Ascending dengan metode Bubble Sort
Jumlah perbandingan data: 88504
Jumlah pertukaran data: 27869
Waktu pengurutan: 0,006673 detik

Mengurutkan Product berdasarkan nama secara Ascending dengan metode Bubble Sort
Jumlah perbandingan data: 86710
Jumlah pertukaran data: 22874
Waktu pengurutan: 0,001502 detik

Mengurutkan Product berdasarkan rating secara Ascending dengan metode Bubble Sort
Jumlah perbandingan data: 72657
Jumlah pertukaran data: 14713
Waktu pengurutan: 0,000875 detik

=====
SELECTION ASC SORT
=====
Mengurutkan Product berdasarkan harga secara Ascending dengan metode Selection Sort
Jumlah perbandingan data: 44850
Jumlah pertukaran data: 300
Waktu pengurutan: 0,001425 detik

Mengurutkan Product berdasarkan nama secara Ascending dengan metode Selection Sort
Jumlah perbandingan data: 44850
Jumlah pertukaran data: 300
Waktu pengurutan: 0,000512 detik

Mengurutkan Product berdasarkan rating secara Ascending dengan metode Selection Sort
Jumlah perbandingan data: 44850
Jumlah pertukaran data: 300
Waktu pengurutan: 0,000556 detik
```

```

=====
BUBBLE DESC SORT
=====
Mengurutkan Product berdasarkan harga secara Descending dengan metode Bubble Sort
Jumlah perbandingan data: 71760
Jumlah pertukaran data: 16173
Waktu pengurutan: 0,007275 detik

Mengurutkan Product berdasarkan nama secara Descending dengan metode Bubble Sort
Jumlah perbandingan data: 84019
Jumlah pertukaran data: 21976
Waktu pengurutan: 0,007869 detik

Mengurutkan Product berdasarkan rating secara Descending dengan metode Bubble Sort
Jumlah perbandingan data: 85215
Jumlah pertukaran data: 20596
Waktu pengurutan: 0,002645 detik

=====
SELECTION DESC SORT
=====
Mengurutkan Product berdasarkan harga secara Descending dengan metode Selection Sort
Jumlah perbandingan data: 44850
Jumlah pertukaran data: 300
Waktu pengurutan: 0,001929 detik

Mengurutkan Product berdasarkan nama secara Descending dengan metode Selection Sort
Jumlah perbandingan data: 44850
Jumlah pertukaran data: 300
Waktu pengurutan: 0,000551 detik

Mengurutkan Product berdasarkan rating secara Descending dengan metode Selection Sort
Jumlah perbandingan data: 44850
Jumlah pertukaran data: 300
Waktu pengurutan: 0,000666 detik

```

Use the output above to analyze the question below:

- **Analyze the results of the sorting and searching that has been done, comparing the results of each sorting and searching algorithm. Which algorithm is the most effective and efficient according to the available data for each sorting and searching?**
- **Mention all the reasons you know about why an algorithm faster or slower than the other!**

Rules:

- **Sorting methods bubble sort and selection sort**
- **Searching methods linear search and binary search**
- **The data compared is 100 data, 200 data, and 300 data**
- **Use data that is still random (Don't use data that already sorted).**