

## MODUL 3

### HARD

Roman is a sports analyst. He works collecting NBA player data and drawing conclusions from that data. On his first day of work, he was given data on 3 NBA teams.

TEAM NAME	GP	W	L	PPG	APG	RPG
Boston Celtics	82	64	18	120.57	26.84	44.60
Golden State Warriors	82	46	36	117.80	29.10	46.89
Los Angeles Lakers	82	47	35	118.04	28.39	43.10

Then, he was assigned to sort the teams. Roman felt that the task was too easy, so to add challenge, he used quicksort to sort them based on the number of team wins.

Diurutkan Berdasarkan Jumlah Kemenangan						
TEAM NAME	GP	W	L	PPG	APG	RPG
Boston Celtics	82	64	18	120.57	26.84	44.60
Los Angeles Lakers	82	47	35	118.04	28.39	43.10
Golden State Warriors	82	46	36	117.80	29.10	46.89

Because the task he was doing was too easy, he asked for another task to work on. For the next task, he was given data on all the players from each team.

Boston Celtics					
PLAYER NAME	JERSEY #	PPG	APG	RPG	
Jayson Tatum	0	23.3	3.6	7.5	
Jaylen Brown	7	17.6	2.7	5.2	
Kristaps Porzingis	8	19.2	1.7	8.9	
Derrick White	9	11.7	4.6	3.8	
Jrue Holiday	4	15.4	6.2	4.9	
Al Horford	42	13.6	3.4	8.1	
Payton Pritchard	11	7.1	1.8	2.0	
Sam Hauser	30	5.7	0.7	2.3	
Luke Kornet	40	5.5	1.0	3.4	
Xavier Tillman	26	5.7	1.2	4.0	
Los Angeles Lakers					
PLAYER NAME	JERSEY #	PPG	APG	RPG	
LeBron James	23	26.8	7.4	7.4	
Anthony Davis	3	24.9	3.8	11.5	
Austin Reaves	15	10.6	2.9	2.8	
Rui Hachimura	28	12.0	1.8	5.3	
D'Angelo Russell	1	18.2	5.7	3.4	
Jarred Vanderbilt	2	6.7	1.4	6.9	
Gabe Vincent	7	7.4	2.4	1.8	
Max Christie	10	3.1	0.6	1.7	
Jaxson Hayes	11	7.5	0.7	4.1	
Christian Wood	35	14.0	1.7	7.1	
Golden State Warriors					
PLAYER NAME	JERSEY #	PPG	APG	RPG	
Stephen Curry	30	26.0	5.4	6.7	
Klay Thompson	11	19.4	2.3	3.9	
Draymond Green	23	9.2	6.8	7.5	
Andrew Higgins	22	18.3	2.5	4.5	
Jonathan Kuminga	0	11.6	2.0	5.2	
Brandin Podziemski	2	9.5	3.8	5.8	
Trayce Jackson-Davis	32	8.0	1.2	5.7	
Moses Moody	4	6.7	1.3	2.9	
Kevon Looney	5	5.6	1.9	6.4	
Gary Payton II	1	5.5	1.4	3.1	

Roman was tasked with finding the NBA MVP candidate. Since the previous task was still too easy for Roman, he decided to use merge sort to sort all players from each team based on PPG (Points Per Game).

Diurutkan Berdasarkan PPG (Points Per Game)				
Boston Celtics				
PLAYER NAME	JERSEY #	PPG	APG	RPG
Jayson Tatum	0	23.3	3.6	7.5
Jaylen Brown	7	17.6	2.7	5.2
Kristaps Porzingis	8	19.2	1.7	8.0
Derrick White	9	11.7	4.6	3.8
Jrue Holiday	4	15.4	6.2	4.0
Al Horford	42	13.6	3.4	8.1
Payton Pritchard	11	7.1	1.8	2.0
Sam Hauser	30	5.7	0.7	2.3
Luke Kornet	40	5.5	1.0	3.4
Xavier Tillman	26	5.7	1.2	4.0
Los Angeles Lakers				
PLAYER NAME	JERSEY #	PPG	APG	RPG
LeBron James	23	26.8	7.4	7.4
Anthony Davis	3	24.9	3.8	11.5
Austin Reaves	15	10.6	2.9	2.8
Rui Hachimura	28	12.0	1.8	5.3
D'Angelo Russell	1	18.2	5.7	3.4
Jarred Vanderbilt	2	6.7	1.4	6.9
Gabe Vincent	7	7.4	2.4	1.8
Max Christie	10	3.1	0.6	1.7
Jaxson Hayes	11	7.5	0.7	4.1
Christian Wood	35	14.0	1.7	7.1
Golden State Warriors				
PLAYER NAME	JERSEY #	PPG	APG	RPG
Stephen Curry	30	26.0	5.4	6.7
Klay Thompson	11	19.4	2.3	3.9
Draymond Green	23	9.2	6.8	7.5
Andrew Wiggins	22	18.3	2.5	4.5
Jonathan Kuminga	0	11.6	2.0	5.2
Brandin Podziemski	2	9.5	3.8	5.8
Trayce Jackson-Davis	32	8.0	1.2	5.7
Moses Moody	4	6.7	1.3	2.9
Kevon Looney	5	5.6	1.9	6.4
Gary Payton II	1	5.5	1.4	3.1

After sorting all players from all teams, Roman felt this method was ineffective for finding the MVP, so he then tried combining all players into one list.

PLAYER NAME	JERSEY #	PPG	APG	RPG
Jayson Tatum	0	23.3	3.6	7.5
Jaylen Brown	7	17.6	2.7	5.2
Kristaps Porzingis	8	19.2	1.7	8.0
Derrick White	9	11.7	4.6	3.8
Jrue Holiday	4	15.4	6.2	4.0
Al Horford	42	13.6	3.4	8.1
Payton Pritchard	11	7.1	1.8	2.0
Sam Hauser	30	5.7	0.7	2.3
Luke Kornet	40	5.5	1.0	3.4
Xavier Tillman	26	5.7	1.2	4.0
LeBron James	23	26.8	7.4	7.4
Anthony Davis	3	24.9	3.8	11.5
Austin Reaves	15	10.6	2.9	2.8
Rui Hachimura	28	12.0	1.8	5.3
D'Angelo Russell	1	18.2	5.7	3.4
Jarred Vanderbilt	2	6.7	1.4	6.9
Gabe Vincent	7	7.4	2.4	1.8
Max Christie	10	3.1	0.6	1.7
Jaxson Hayes	11	7.5	0.7	4.1
Christian Wood	35	14.0	1.7	7.1
Stephen Curry	30	26.0	5.4	6.7
Klay Thompson	11	19.4	2.3	3.9
Draymond Green	23	9.2	6.8	7.5
Andrew Wiggins	22	18.3	2.5	4.5
Jonathan Kuminga	0	11.6	2.0	5.2
Brandin Podziemski	2	9.5	3.8	5.8
Trayce Jackson-Davis	32	8.0	1.2	5.7
Moses Moody	4	6.7	1.3	2.9
Kevon Looney	5	5.6	1.9	6.4
Gary Payton II	1	5.5	1.4	3.1

After creating a list of all players in the NBA, Roman sorted all players based on PPG using merge sort.

Semua Pemain Diurutkan Berdasarkan PPG (Points Per Game) Menggunakan Merge Sort				
PLAYER NAME	JERSEY #	PPG	APG	RPG
LeBron James	23	26.8	7.4	7.4
Stephen Curry	30	26.0	5.4	6.7
Anthony Davis	3	24.9	3.8	11.5
Jayson Tatum	0	23.3	3.6	7.5
Klay Thompson	11	19.4	2.3	3.9
Kristaps Porzingis	8	19.2	1.7	8.0
Andrew Wiggins	22	18.3	2.5	4.5
D'Angelo Russell	1	18.2	5.7	3.4
Jaylen Brown	7	17.6	2.7	5.2
Jrue Holiday	4	15.4	6.2	4.0
Christian Wood	35	14.0	1.7	7.1
Al Horford	42	13.6	3.4	8.1
Rui Hachimura	28	12.0	1.8	5.3
Derrick White	9	11.7	4.6	3.8
Jonathan Kuminga	0	11.6	2.0	5.2
Austin Reaves	15	10.6	2.9	2.8
Brandin Podziemski	2	9.5	3.8	5.8
Draymond Green	23	9.2	6.8	7.5
Trayce Jackson-Davis	32	8.0	1.2	5.7
Jaxson Hayes	11	7.5	0.7	4.1
Gabe Vincent	7	7.4	2.4	1.8
Payton Pritchard	11	7.1	1.8	2.0
Moses Moody	4	6.7	1.3	2.9
Jarred Vanderbilt	2	6.7	1.4	6.9
Xavier Tillman	26	5.7	1.2	4.0
Sam Hauser	30	5.7	0.7	2.3
Kevon Looney	5	5.6	1.9	6.4
Gary Payton II	1	5.5	1.4	3.1
Luke Kornet	40	5.5	1.0	3.4
Max Christie	10	3.1	0.6	1.7
Elapsed Time is 0.0739 msec				

Since Roman had learned quick sort the previous day, he tried using it for this task.

Semua Pemain Diurutkan Berdasarkan PPG (Points Per Game) Menggunakan Quick Sort				
PLAYER NAME	JERSEY #	PPG	APG	RPG
LeBron James	23	26.8	7.4	7.4
Stephen Curry	30	26.0	5.4	6.7
Anthony Davis	3	24.9	3.8	11.5
Jayson Tatum	0	23.3	3.6	7.5
Klay Thompson	11	19.4	2.3	3.9
Kristaps Porzingis	8	19.2	1.7	8.0
Andrew Wiggins	22	18.3	2.5	4.5
D'Angelo Russell	1	18.2	5.7	3.4
Jaylen Brown	7	17.6	2.7	5.2
Jrue Holiday	4	15.4	6.2	4.0
Christian Wood	35	14.0	1.7	7.1
Al Horford	42	13.6	3.4	8.1
Rui Hachimura	28	12.0	1.8	5.3
Derrick White	9	11.7	4.6	3.8
Jonathan Kuminga	0	11.6	2.0	5.2
Austin Reaves	15	10.6	2.9	2.8
Brandin Podziemski	2	9.5	3.8	5.8
Draymond Green	23	9.2	6.8	7.5
Trayce Jackson-Davis	32	8.0	1.2	5.7
Jaxson Hayes	11	7.5	0.7	4.1
Gabe Vincent	7	7.4	2.4	1.8
Payton Pritchard	11	7.1	1.8	2.0
Moses Moody	4	6.7	1.3	2.9
Jarred Vanderbilt	2	6.7	1.4	6.9
Xavier Tillman	26	5.7	1.2	4.0
Sam Hauser	30	5.7	0.7	2.3
Kevon Looney	5	5.6	1.9	6.4
Gary Payton II	1	5.5	1.4	3.1
Luke Kornet	40	5.5	1.0	3.4
Max Christie	10	3.1	0.6	1.7
Elapsed Time is 0.0535 msec				

Then Roman wanted to find players based on their PPG, so he used linear search and binary search to search through the sorted player list (regardless of whether it was merged or quick-sorted).

Mencari Pemain dengan PPG 23.3 Menggunakan Linear Search  
Ditemukan Pemain: Jayson Tatum  
Elapsed Time is 0.0106 msec

Mencari Pemain dengan PPG 23.3 Menggunakan Binary Search  
Ditemukan Pemain: Jayson Tatum  
Elapsed Time is 0.0187 msec

Since Roman wanted to find the best player so far, he combined linear search and binary search (by first finding the player's PPG using binary search, then using linear search for upward iteration) to find MVP candidates based on minimum PPG.

Mencari Kandidat MVP berdasarkan PPG $\geq$ 23.3				
PLAYER NAME	JERSEY #	PPG	APG	RPG
LeBron James	23	26.8	7.4	7.4
Stephen Curry	30	26.0	5.4	6.7
Anthony Davis	3	24.9	3.8	11.5
Jayson Tatum	0	23.3	3.6	7.5

## Notes :

- To calculate the search and sort time, you can use the following code or another algorithm:

```
long startTime = System.nanoTime();  
//kode  
long elapsedTime = System.nanoTime() - startTime;  
System.out.println("Elapsed Time is " + (elapsedTime /  
1000000.0) + " msec");
```

## Aturan :

- Use the merge sort and selection sort methods.
- Use the linear search and binary search methods.
- Perform sorting and searching on the data and compare the time effectiveness of each sorting and searching. (The data compared is 100 data, 200 data and 300 data).
- Example of comparison results

Number of Data	Linear Search	Binary Search	Merge Sort	Quick Sort
100	time	time	time	time
200	time	time	time	time
300	time	time	time	time

- Analyze the results of the sorting and searching that has been done, compare 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.