PROGRESS OUTCOME 6

Linear versus binary searches

Context

Sharee has been investigating the impact of using different algorithms for searching and sorting data. She has experimented with two algorithms to determine which one works better for searching different data sets.

Insight 1: Applications of searching

Most people don't realise how much they rely on computers use of fast and accurate search algorithms for all kinds of information. So I researched and brainstormed the different types of everyday computer searches. We take it for granted that a web search will provide instant results and that our groceries will scan quickly at the supermarket checkout. In medicine, doctors and researchers need to search databases of diseases or genetic information. The police need quick access to information about offenders and public safety. Because of the computer's ability to quickly search large amounts of data for answers to problems, artificial intelligence is developing rapidly.

Insight 2: Different algorithms for searching data

During my research, I discovered many different search algorithms. Some are kept confidential, like the Google™ search algorithm, to maintain their competitive edge over other businesses. However, linear and binary searches are two basic search algorithms that most computer science students are familiar with.

To understand how they work, I tried each one with a friend. First, I printed out fifty random numbers and put them face down in random order. I then asked my friend to find a specific number using a linear search. That meant he had to start with the first number and look at each one in order until he found the number. We worked out that it could take fifty attempts to find the correct number, if it was the last one.

I then put the numbers in numerical order and asked him to find the number using a binary search by cutting the data set in half each time. He started in the middle and identified whether the specific number was higher or lower than the middle number and therefore which half of the data set it was in. We worked out that using this approach would take up to six attempts to find the number, because after cutting the data set in half six times you have only one piece of data left.

Insight 3: Determining best, average and worst cases

After doing the experiment with my friend, I used a spreadsheet to look at the best, average, and worst cases when searching data using these two algorithms. I also researched this on the internet to see if my conclusion was correct.

I learnt that the worst case for a linear search is that you must look through every piece of data, whereas a binary search cuts the data in half each time, so at worst you have to look through the data as many times as you can cut it in half (i.e., until there's one item left).

On average, that means it's much quicker for a computer to perform a binary search.
For example, if I have 1024 items of data to search through, it will take me at most 10 searches:

1) 1024/2 = 512

2) 512/2 = 256

3) 256/2 = 128

...

9) 4/2 = 2

10) 2/2 = 1

From my research, I learnt that this is a log base 2 function, so I used the log base 2 function in my spreadsheet to work out the worst case. I also read that the average and the worst case for a binary search are just about the same.

I tested this by downloading and running a binary search program 20 times for each data set and put the results into a spreadsheet. I saw that the average and worst cases were almost exactly the same. So I used the same spreadsheet formula for the average as for the worst case scenario (log base 2).

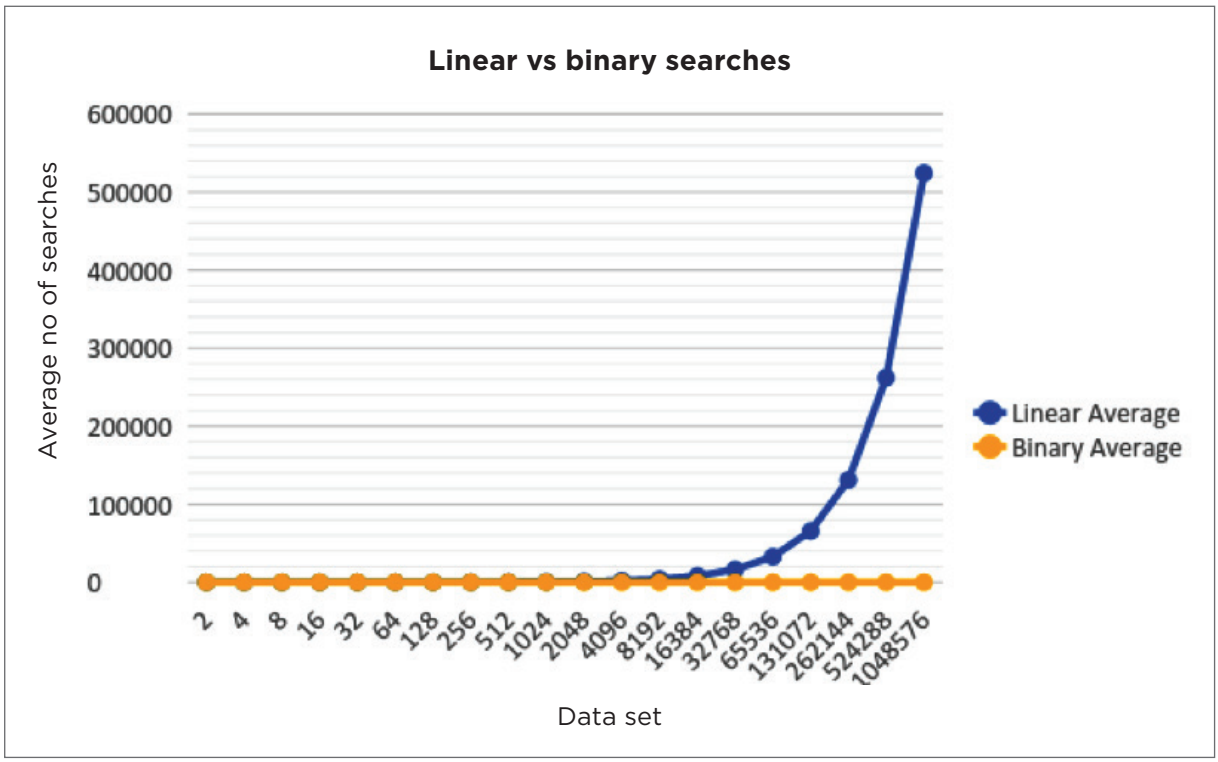
The table below shows my conclusions:

Linear Search			
Data set	Best	Worst	Average
2	1	1	1
4	1	3	2
8	1	7	4
16	1	15	8
32	1	31	16
64	1	63	32
128	1	127	64
256	1	255	128
512	1	511	256
1024	1	1023	512

Binary Search			
Data set	Best	Worst	Average
2	1	1	1
4	1	2	2
8	1	3	3
16	1	4	4
32	1	5	5
64	1	6	6
128	1	7	7
256	1	8	8
512	1	9	9
1024	1	10	10

Insight 4: Cost changes as problem size increases

I then added further data sets to the spreadsheet and made a graph comparing the two searches. I concluded that using a linear search with large amounts of data is much slower than a binary search because each time the data set doubles, the average number of searches for a linear search doubles, whereas it only adds one more search for a binary search.



ISBN: 978-1-77669-236-1

MINISTRY OF EDUCATION
TE TĀHUHU O TE MĀTAURANGA