Compulsory 2

Github: https://github.com/Jonathannygaard/Compulsory2.git

I have chosen Bubblesort, quicksort and merge sort.

Bubblesort: In bubblesort you compare every number with the next swapping them if the right number is smaller than the left number. This check is performed (size of vector)^2 times making it a simple sorting algorithm, but pretty slow.

Quicksort: In quicksort you choose a pivot point and move all smaller values to the left of the pivot and all larger values to the right. After this you sort each side of the pivot, you choose a new pivot within each side and sort them using this.

Mergesort: In mergesort you divide the vector until you have base cases in every part, this means only one value. It then combines the part back together in order.

Time complexity:

Bubblesort can be the most time efficient if you have the best case. But if you have anything other then the best case it quickly gets slower as the average and worst case is n^2. Space complexity is fixed using the same amount of memory regardless of size and case.

Quicksort is more reliable than bubblesort as the average time is n(log(n)) instead of bubblesort's n^2. Quicksort is slower than bubble sort at best case and they are the same at worst case. Quicksort would use more memory than bubblesort as quicksort uses log(n) memory compared to bubblesort's 1 memory use.

Mergesort is quicker than quicksort and bubblesort in all instances other then best case where bubblesort is quicker. Merge sort uses nlog(n) time at all cases. Merge sort however uses more memory as it uses n memory.

Time:

Algorithm	10	100	1000	10000
Bubblesort	0,0081 ms	0,1257 ms	13,8166 ms	1071,87 ms
Quicksort	0,002 ms	0,0209 ms	0,2623 ms	2,8235 ms
Mergesort	0,0358 ms	0,1259 ms	2,388 ms	14,1398 ms

I think the quicksort algorithm is the best because it’s one of the quickest as well as not using too much memory. It also performed better than mergesort on time even though it should have been as fast or faster. This is probably because of poor implementation from me. I also feel like quicksort is easier to implement.