COURSE NAME: ALGORITHM ANALYSIS

COURSE INSTRUCTOR: Dr. M. Amaç GÜVENSAN

STUDENT NAME: Ertuğrul ŞENTÜRK

STUDENT NO: 18011028

STUDENT E-MAIL: mdesenturk@gmail.com

SCHOOL YEAR: 3

GROUP NUMBER: 1

HOMEWORK NUMBER: 1

QUESTION NUMBER: 1

QUESTION: Find two closest number in an unordered array

ALGORITHM:

a-) In question a for a brute-force solution;

- Compared all the elements of the array in pairs.
- The absolute value of the difference between two elements is calculated.
- Kept the indices of the two elements with the smallest difference in absolute value, and then these elements printed as an output.

b-) For the optimal solution in question b;

- Array is sorted with merge sort algorithm.
- The elements of the listed array are compared with the next element in order.
- The elements with the lowest difference are outputted to the user.

ALGORITHM ANALYSIS:

a-) For brute force solution in case A;

Since there are 2 nested for loops, we choose the most repetitive part of the algorithm as the "if (diff <min)" comparison within the for loops. Since this comparison is done (n * n) / 2 times in all cases, the complexities of worst, best and average case are equal. In this case, too, the complexity is $\theta(n^2)$.

b-) For the optimal solution in case B;

In this algorithm, we navigate the merge sort algorithm and then the array with just a loop. Since the complexity of the sort algorithm will be higher and the merge sort algorithm is a divide and conquer algorithm, we can apply the master's theorem.

Since the most complex place in this algorithm is the merge part, if we take the comparison in the loop as the most frequently repeated event:

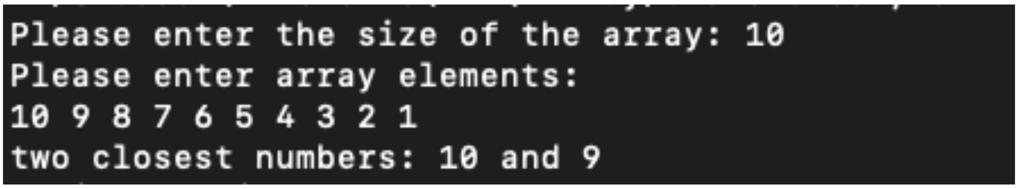
$C_{worst}(n) = 2 * C_{worst}(n / 2) + n-1$; when $n> 1$ and $C(1) = 0$.

From here, according to the master theorem, $a = 2$, $b = 2$, $d = 1$. According to this result, since $a = b^d$, the complexity is calculated as $\theta(n * \log (n))$.

PROGRAM OUTPUTS:

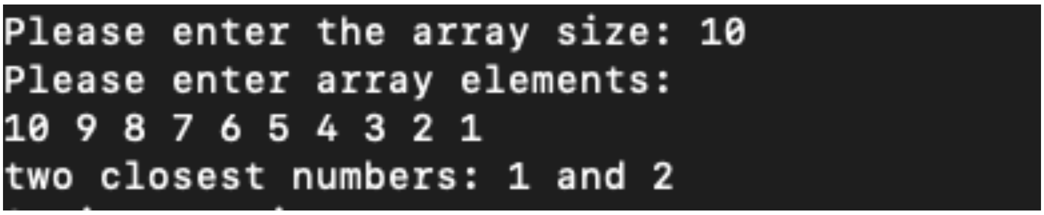
1-) I used numbers from 10 to 1 as input.

a)



```
Please enter the size of the array: 10
Please enter array elements:
10 9 8 7 6 5 4 3 2 1
two closest numbers: 10 and 9
```

b)

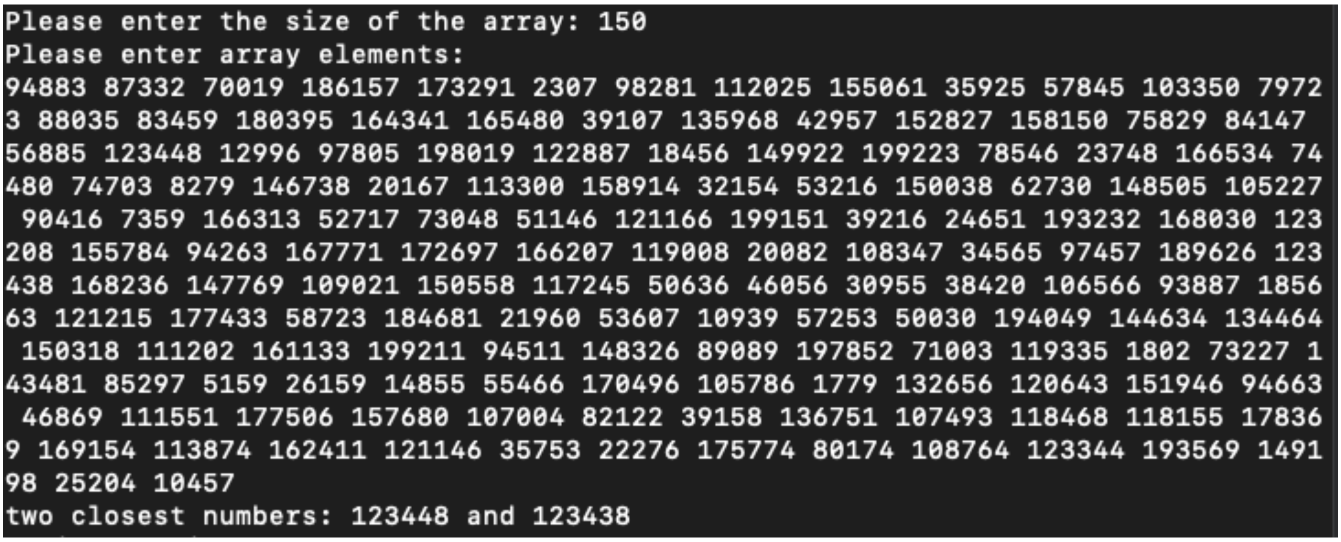


```
Please enter the array size: 10
Please enter array elements:
10 9 8 7 6 5 4 3 2 1
two closest numbers: 1 and 2
```

The main difference between the two algorithms is if there is more than one case in which the difference between the two arrays is the smallest, algorithm 1 gives the first two smallest elements according to the given order of the array. As the second algorithm ranks the array, it gives the numbers with the smallest difference and value.

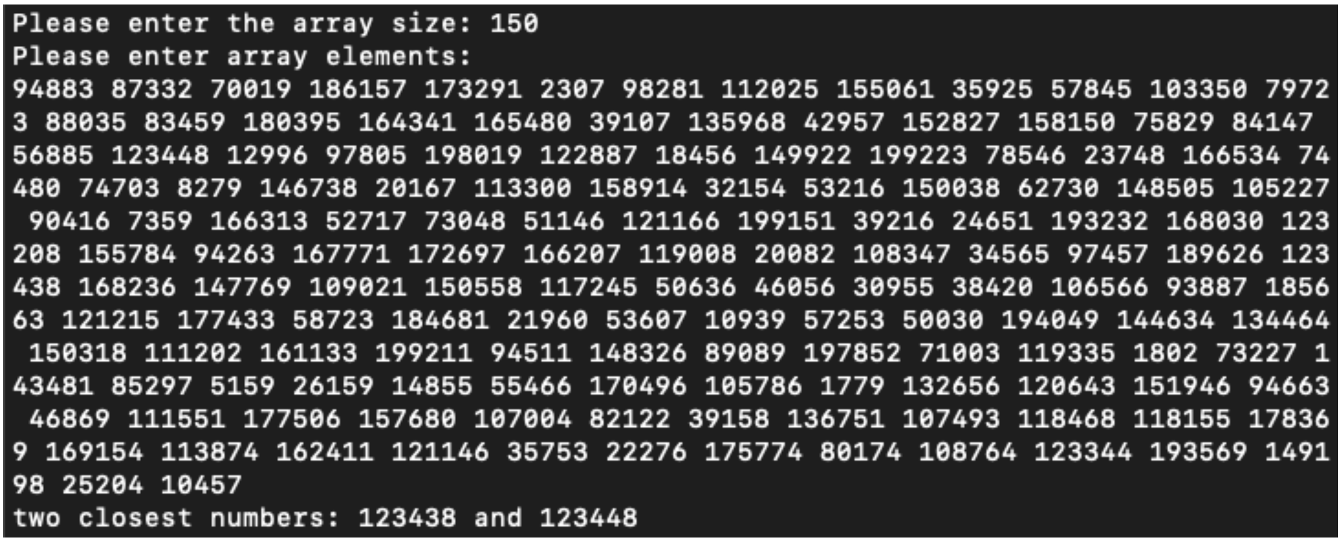
2-) I used a 150-element array with numbers between 0 and 200,000 as input.

a)



```
Please enter the size of the array: 150
Please enter array elements:
94883 87332 70019 186157 173291 2307 98281 112025 155061 35925 57845 103350 7972
3 88035 83459 180395 164341 165480 39107 135968 42957 152827 158150 75829 84147
56885 123448 12996 97805 198019 122887 18456 149922 199223 78546 23748 166534 74
480 74703 8279 146738 20167 113300 158914 32154 53216 150038 62730 148505 105227
 90416 7359 166313 52717 73048 51146 121166 199151 39216 24651 193232 168030 123
208 155784 94263 167771 172697 166207 119008 20082 108347 34565 97457 189626 123
438 168236 147769 109021 150558 117245 50636 46056 30955 38420 106566 93887 1856
63 121215 177433 58723 184681 21960 53607 10939 57253 50030 194049 144634 134464
 150318 111202 161133 199211 94511 148326 89089 197852 71003 119335 1802 73227 1
43481 85297 5159 26159 14855 55466 170496 105786 1779 132656 120643 151946 94663
 46869 111551 177506 157680 107004 82122 39158 136751 107493 118468 118155 17836
9 169154 113874 162411 121146 35753 22276 175774 80174 108764 123344 193569 1491
98 25204 10457
two closest numbers: 123448 and 123438
```

b)



```
Please enter the array size: 150
Please enter array elements:
94883 87332 70019 186157 173291 2307 98281 112025 155061 35925 57845 103350 7972
3 88035 83459 180395 164341 165480 39107 135968 42957 152827 158150 75829 84147
56885 123448 12996 97805 198019 122887 18456 149922 199223 78546 23748 166534 74
480 74703 8279 146738 20167 113300 158914 32154 53216 150038 62730 148505 105227
 90416 7359 166313 52717 73048 51146 121166 199151 39216 24651 193232 168030 123
208 155784 94263 167771 172697 166207 119008 20082 108347 34565 97457 189626 123
438 168236 147769 109021 150558 117245 50636 46056 30955 38420 106566 93887 1856
63 121215 177433 58723 184681 21960 53607 10939 57253 50030 194049 144634 134464
 150318 111202 161133 199211 94511 148326 89089 197852 71003 119335 1802 73227 1
43481 85297 5159 26159 14855 55466 170496 105786 1779 132656 120643 151946 94663
 46869 111551 177506 157680 107004 82122 39158 136751 107493 118468 118155 17836
9 169154 113874 162411 121146 35753 22276 175774 80174 108764 123344 193569 1491
98 25204 10457
two closest numbers: 123438 and 123448
```