Name: Ngo Thi Hong Ly
ID: 22520836
Class: CS4343.P11.CTTT.1
Github: https://github.com/prettyfairyy/DSA---Lab1.git

DATA STRUCTURES & ALGORITHMS
Lab 4: Sorting algorithm

Question 1:

```
Sorting 1K integers with different algorithms:
Bubble Sort time: 0.0021528 seconds
Selection Sort time: 0.000563 seconds
Insertion Sort time: 0.0004971 seconds
Quick Sort time: 9.1e-05 seconds
PS C:\Users\lyngo>
```

Question 2:

```
Bubble Sort time for size 100: 2.61e-05 seconds
Selection Sort time for size 100: 9.9e-06 seconds
Insertion Sort time for size 100: 4.6e-06 seconds
Quick Sort time for size 100: 5.6e-06 seconds
Merge Sort time for size 100: 7.6e-06 seconds
-----------------------------------
Bubble Sort time for size 1000: 0.0020419 seconds
Selection Sort time for size 1000: 0.0004943 seconds
Insertion Sort time for size 1000: 0.0003447 seconds
Quick Sort time for size 1000: 0.0001173 seconds
Merge Sort time for size 1000: 0.0001086 seconds
-----------------------------------
Bubble Sort time for size 10000: 0.243688 seconds
Selection Sort time for size 10000: 0.0681895 seconds
Insertion Sort time for size 10000: 0.047936 seconds
Quick Sort time for size 10000: 0.0012755 seconds
Merge Sort time for size 10000: 0.0019321 seconds
-----------------------------------
```

Question 3:

```
Sorting integers:
Bubble Sort: 1 2 3 4 5 6 7 8
Time: 7e-07 seconds
Selection Sort: 1 2 3 4 5 6 7 8
Time: 7e-07 seconds
Insertion Sort: 1 2 3 4 5 6 7 8
Time: 3e-07 seconds
Quick Sort: 1 2 3 4 5 6 7 8
Time: 7e-07 seconds
Merge Sort: 1 2 3 4 5 6 7 8
Time: 2.4e-06 seconds
Sorting floating-point numbers:
Bubble Sort: 1.1 2.2 3.3 4.4 5.5 6.6 7.7 8.8
Time: 6e-07 seconds
Selection Sort: 1.1 2.2 3.3 4.4 5.5 6.6 7.7 8.8
Time: 5e-07 seconds
Insertion Sort: 1.1 2.2 3.3 4.4 5.5 6.6 7.7 8.8
Time: 3e-07 seconds
Quick Sort: 1.1 2.2 3.3 4.4 5.5 6.6 7.7 8.8
Time: 8e-07 seconds
Merge Sort: 1.1 2.2 3.3 4.4 5.5 6.6 7.7 8.8
Time: 2.8e-06 seconds
Sorting strings:
Bubble Sort: apple banana grape kiwi orange pear
Time: 1e-06 seconds
Selection Sort: apple banana grape kiwi orange pear
Time: 1e-06 seconds
Insertion Sort: apple banana grape kiwi orange pear
Time: 9e-07 seconds
Quick Sort: apple banana grape kiwi orange pear
Time: 2.6e-06 seconds
Merge Sort: apple banana grape kiwi orange pear
Time: 2.5e-06 seconds
PS C:\Users\lyngo>
```

Question 4:

```
Linked List before sorting: 0 2 6 5 8 7 9 3 1 4
Linked List sorting time: 1.2e-06 seconds
Linked List after sorting: 0 1 2 3 4 5 6 7 8 9
---------------------------------
Queue before sorting: 4 1 3 9 7 8 5 6 2 0
Queue sorting time: 1.7e-05 seconds
Queue after sorting: 1 4 3 9 7 8 5 6 2 0
---------------------------------
Stack before sorting: 4 1 3 9 7 8 5 6 2 0
Stack sorting time: 2.1e-05 seconds
Stack after sorting: 0 1 2 3 4 5 6 7 8 9
---------------------------------
PS C:\Users\lyngo>
```

Question 5:

```
Unsorted array: 4144 7598 7573 6814 3431 5883 5861 2053 3263 9586
Standard Insertion Sort time: 0.0004617 seconds
Binary Insertion Sort time: 0.0002899 seconds
PS C:\Users\lyngo>
```

Question 6:

```
Original array: 3 9 2 1 4 5
Building max-heap: heapifying index 2
Heapifying at index 2, left = 5, right = 6
Left child 5 is larger than current largest 2
Swapping 2 with 5
Heapifying at index 5, left = 11, right = 12
Building max-heap: heapifying index 1
Heapifying at index 1, left = 3, right = 4
Building max-heap: heapifying index 0
Heapifying at index 0, left = 1, right = 2
Left child 9 is larger than current largest 3
Swapping 3 with 9
Heapifying at index 1, left = 3, right = 4
Right child 4 is larger than current largest 3
Swapping 3 with 4
Heapifying at index 4, left = 9, right = 10
Max-heap built successfully.
Maximum element found at the root of the heap: 9
Array after building max-heap (structure only, not fully sorted): 9 4 5 1 3 2
Maximum element in the array is: 9
PS C:\Users\lyngo>
```

Question 7:

```
Unsorted array (first 10 elements): 5489 2261 4811 2196 3196 5296 239 8431 3063 9718
Standard Merge Sort time: 9.18e-05 seconds
Hybrid Merge-Insertion Sort time: 6.76e-05 seconds
PS C:\Users\lyngo>
```

Question 8:

```
Unsorted array (first 10 elements): 5907 1801 433 2108 2741 6397 2648 6956 6626 9197
Standard Quick Sort time: 7.52e-05 seconds
Hybrid Quick-Insertion Sort time: 6e-05 seconds
PS C:\Users\lyngo>
```

Question 9:

```
Original points:
(3, 4) (1, 1) (0, 0) (5, 12) (9, 10) (8, 15) (2, 3) (7, 24) (6, 8) (4, 9)

Sorting Points by distance from origin:
Bubble Sort took 1.2e-06 seconds
Sorted points by Bubble Sort: (0, 0) (1, 1) (2, 3) (3, 4) (4, 9) (6, 8) (5, 12) (9, 10) (8, 15) (7, 24)

Selection Sort took 1e-06 seconds
Sorted points by Selection Sort: (0, 0) (1, 1) (2, 3) (3, 4) (4, 9) (6, 8) (5, 12) (9, 10) (8, 15) (7, 24)

Insertion Sort took 7e-07 seconds
Sorted points by Insertion Sort: (0, 0) (1, 1) (2, 3) (3, 4) (4, 9) (6, 8) (5, 12) (9, 10) (8, 15) (7, 24)

Quick Sort took 8e-07 seconds
Sorted points by Quick Sort: (0, 0) (1, 1) (2, 3) (3, 4) (4, 9) (6, 8) (5, 12) (9, 10) (8, 15) (7, 24)
PS C:\Users\lyngo>
```

Homework:

```
Sorting Exercises Menu:
1. Sort Products (E-commerce website)
2. Sort Social Media Posts
3. Sort Students (School Management)
4. Sort Songs (Music Playlist)
0. Exit
Enter your choice: 2
Posts sorted by likes:
Content: Post 8, Likes: 7
Content: Post 9, Likes: 17
Content: Post 10, Likes: 24
Content: Post 6, Likes: 39
Content: Post 5, Likes: 42
Content: Post 2, Likes: 44
Content: Post 1, Likes: 49
Content: Post 3, Likes: 67
Content: Post 4, Likes: 82
Content: Post 7, Likes: 82

Sorting Exercises Menu:
1. Sort Products (E-commerce website)
2. Sort Social Media Posts
3. Sort Students (School Management)
4. Sort Songs (Music Playlist)
0. Exit
Enter your choice: 4
Songs sorted by mood:
Title: Song 5, Mood: 0
Title: Song 3, Mood: 1
Title: Song 6, Mood: 1
Title: Song 4, Mood: 2
Title: Song 7, Mood: 2
Title: Song 8, Mood: 3
Title: Song 1, Mood: 5
Title: Song 2, Mood: 7
Title: Song 9, Mood: 7
Title: Song 10, Mood: 7
```

```
Sorting Exercises Menu:
1. Sort Products (E-commerce website)
2. Sort Social Media Posts
3. Sort Students (School Management)
4. Sort Songs (Music Playlist)
0. Exit
Enter your choice: 1
Products sorted by price:
Name: Product3, Price: $4
Name: Product9, Price: $13
Name: Product1, Price: $27
Name: Product6, Price: $34
Name: Product4, Price: $55
Name: Product2, Price: $56
Name: Product7, Price: $60
Name: Product8, Price: $72
Name: Product10, Price: $89
Name: Product5, Price: $98

Sorting Exercises Menu:
1. Sort Products (E-commerce website)
2. Sort Social Media Posts
3. Sort Students (School Management)
4. Sort Songs (Music Playlist)
0. Exit
Enter your choice: 3
Students sorted by grade:
Name: Student 4, Grade: 8
Name: Student 6, Grade: 8
Name: Student 1, Grade: 22
Name: Student 8, Grade: 24
Name: Student 9, Grade: 26
Name: Student 5, Grade: 32
Name: Student 10, Grade: 50
Name: Student 3, Grade: 86
Name: Student 7, Grade: 86
Name: Student 2, Grade: 97
```