

Assignment 6 — Sorting Algorithm

TA: Kile (a256673@hotmail.com)

Deadline: 2018/01/09 11:59 pm

1. Introduction

Sorting is a common and important problem in programming. To understand the different sorting algorithms, we are going to implement the game named “6 Nimmt!” in this assignment.

To play the game, you shuffle 104 number cards, lay out four cards to start the four rows, and then deal ten cards to each player. For each turn, each player takes turn to discard the first card in the hands, and add the card to the rows with cards being placed in ascending order based on the card number. Furthermore, the card is placed in the row that ends with the highest number. When the fifth card is placed in a row, the owner of that card claims the other four cards and the fifth card becomes the first card in that row. A player who claims one card will lose one point. After finishing ten rounds, players tally their scores. The player with the most points wins!

2. Game rules

- (1) The card is numbered from 1 to 104. No duplicate numbers will appear.
- (2) There are four rows. Every row holds up to five cards.
- (3) For each turn, the player with smaller card's number has higher priority to discard.
- (4) Each player has 66 points initially.
- (5) If the player places the fifth card in a row, he/she will lose 4 points.
- (6) If the player discard the card number which is smaller than all cards in all rows, he/she claims all cards in the row which has fewest cards. However, if every row has the same amount of cards, the player should claim all cards in the row which has the smallest card number.

You are given a file named “test1.txt” as your input. A sample input is shown in Figure 1. The top four numbers are the first card of four rows. The fifth line to the last line of the file will deal to ten players in order.

89
49
75
43
53, 12, 46, 81, 76, 37, 34, 73, 6, 92
19, 24, 104, 79, 30, 71, 101, 70, 32, 72
38, 14, 58, 57, 98, 35, 50, 62, 80, 18
10, 94, 15, 33, 100, 4, 74, 78, 52, 3
26, 95, 27, 86, 96, 61, 68, 64, 11, 8
25, 48, 17, 40, 99, 47, 77, 97, 59, 88
29, 21, 67, 56, 28, 36, 31, 69, 87, 63
54, 39, 103, 55, 83, 16, 93, 1, 102, 7
51, 9, 90, 42, 60, 82, 2, 20, 41, 44
45, 84, 66, 13, 91, 85, 22, 65, 5, 23

Figure 1. test1.txt

For rule #3, you must implement three different sorting algorithms for the specified rounds. you are required to implement quick sort algorithm for round 3, bubble sort algorithm for round 1,2,4,5,6,8 and 9, and selection sort algorithm for round 7. The following illustration shows the running results of the input example in Figure 1.

Round	Illustration																
0	<div><div>89</div><div></div><div></div><div></div><div></div></div> <div><div>49</div><div></div><div></div><div></div><div></div></div> <div><div>75</div><div></div><div></div><div></div><div></div></div> <div><div>43</div><div></div><div></div><div></div><div></div></div>				Player	Cards											Point
					1	53	12	46	81	76	37	34	73	6	92	66	
					2	19	24	104	79	30	71	101	70	32	72	66	
					3	38	14	58	57	98	35	50	62	80	18	66	
					4	10	94	15	33	100	4	74	78	52	3	66	
					5	26	95	27	86	96	61	68	64	11	8	66	
					6	25	48	17	40	99	47	77	97	59	88	66	
					7	29	21	67	56	28	36	31	69	87	63	66	
					8	54	39	103	55	83	16	93	1	102	7	66	
					9	51	9	90	42	60	82	2	20	41	44	66	
					10	45	84	66	13	91	85	22	65	5	23	66	
1	<p>10(Player4)<19(Player2)<25(Player6)<26(Player5)<29(Player7)<38(Player3)<45(Player10)<51(Player9)<53(Player1)<54(Player8) (ATTENTION: In round 3, the above sorting should be done by the quick sort algorithm. In round 1,2,3,4,5,6,8 and 9, the above sorting should be done by the bubble sort algorithm. In round 7, the above sorting should be done by selection sort algorithm.) According to rule #3, player 4 firstly discard. Because player 4 discard 10 that is smaller than all cards in all rows (89,49,75,43), according to rule #6, his/her card becomes the first card in the fourth row. And 1 point is deducted from player 4 (the results are shown below).</p>																

89				
49				
75				
10				

Player	Cards										Point
1	53	12	46	81	76	37	34	73	6	92	66
2	19	24	104	79	30	71	101	70	32	72	66
3	38	14	58	57	98	35	50	62	80	18	66
4		94	15	33	100	4	74	78	52	3	65
5	26	95	27	86	96	61	68	64	11	8	66
6	25	48	17	40	99	47	77	97	59	88	66
7	29	21	67	56	28	36	31	69	87	63	66
8	54	39	103	55	83	16	93	1	102	7	66
9	51	9	90	42	60	82	2	20	41	44	66
10	45	84	66	13	91	85	22	65	5	23	66

After player 4 is done, it is player 2's turn to discard one card (the results are shown below).

89				
49				
75				
10	19			

Player	Cards										Point
1	53	12	46	81	76	37	34	73	6	92	66
2		24	104	79	30	71	101	70	32	72	66
3	38	14	58	57	98	35	50	62	80	18	66
4		94	15	33	100	4	74	78	52	3	65
5	26	95	27	86	96	61	68	64	11	8	66
6	25	48	17	40	99	47	77	97	59	88	66
7	29	21	67	56	28	36	31	69	87	63	66
8	54	39	103	55	83	16	93	1	102	7	66
9	51	9	90	42	60	82	2	20	41	44	66
10	45	84	66	13	91	85	22	65	5	23	66

After player 2 is done, it is player 6's turn to discard one card (the results are shown below).

89				
49				
75				
10	19	25		

Player	Cards										Point
1	53	12	46	81	76	37	34	73	6	92	66
2		24	104	79	30	71	101	70	32	72	66
3	38	14	58	57	98	35	50	62	80	18	66
4		94	15	33	100	4	74	78	52	3	65
5	26	95	27	86	96	61	68	64	11	8	66
6		48	17	40	99	47	77	97	59	88	66
7	29	21	67	56	28	36	31	69	87	63	66
8	54	39	103	55	83	16	93	1	102	7	66
9	51	9	90	42	60	82	2	20	41	44	66
10	45	84	66	13	91	85	22	65	5	23	66

After player 6 is done, it is player 5's turn to discard one card (the results are shown below).

89				
49				
75				
10	19	25	26	

Player	Cards										Point
1	53	12	46	81	76	37	34	73	6	92	66
2		24	104	79	30	71	101	70	32	72	66
3	38	14	58	57	98	35	50	62	80	18	66
4		94	15	33	100	4	74	78	52	3	65
5		95	27	86	96	61	68	64	11	8	66
6		48	17	40	99	47	77	97	59	88	66
7	29	21	67	56	28	36	31	69	87	63	66
8	54	39	103	55	83	16	93	1	102	7	66
9	51	9	90	42	60	82	2	20	41	44	66
10	45	84	66	13	91	85	22	65	5	23	66

After player 5 is done, it is player 7's turn to discard one card. According to rule #5, 4 points are deducted from player 7 (the results are shown below).

[illegible]

ATTENTION: You must output a result.txt that record the conditions in every row, card sorting results, and every player's point in round 3, round 5, round 7, and round 10. Taking round 3 as an example:

```

This is 3 round

This is 1 row : 104

This is 2 row : 15 17 27

This is 3 row : 24 39 46

This is 4 row : 58 66 67 90

PlayerName : 1 Point: 66
PlayerName : 2 Point: 58
PlayerName : 3 Point: 62
PlayerName : 4 Point: 64
PlayerName : 5 Point: 66
PlayerName : 6 Point: 66
PlayerName : 7 Point: 62
PlayerName : 8 Point: 66
PlayerName : 9 Point: 65
PlayerName : 10 Point: 62

```

3. How to submit

To submit your file electronically, enter the following command from the csie workstation:

```
turnin ds.hw6 [your files...]
```

To check the file you turnin, enter the following command from the csie workstation:

```
turnin -ls ds.hw6
```

You can see other description about turnin from the following link:

<https://www.cs.ccu.edu.tw/lab401/doku.php?id=turninhowto>

4. Grade

The TA(s) will mark and give points according to the following grading polices:

- 10% Source code can be compiled without any error
- 10% Readme file, code style, and comments in source code
- 80% Three required sorting algorithms and the result correctness (result.txt)

A readme file should include your name, class ID, a brief description of the code, and other issues students think that will be helpful for the TAs to understand their homework.