**CS 515 Exercise A04: Sorting**

\_\_\_\_\_ /100

**Names: Ryan J. Skelly\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Wildcats ID’s : 988977861\_\_\_\_\_\_\_\_\_\_\_**

**Lecture Section: 01 / 02 (circle one)**

**[50 pts.]** Show how you would use heapsort to sort the following array in ascending order. In the process, be sure to draw a picture of the starting heap structure prior to any movement of elements and the final state of the heap when fully constructed. Also show the heap after each of the first two extractions of the max and show the array in sorted order.

Initial array:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 12 | 9 | 1 | 3 | 5 | 8 | 6 | 15 | 7 | 4 |

15

12 8

9 5 1 6

3 7 4

12

9 1

3 5 8 6

15 7 4

9

7 8

4 5 1 6

3

12

9 8

7 5 1 6

3 4

Final array:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 12 | 15 |

*More on the back…*

**[50 pts.]** Use Quicksort to sort the following array in ascending order. At each step, circle the right-most number in each range as the pivot for that range, and then partition the remaining numbers into the next step (but do not copy the pivot values into the next step—instead, place an X where it should go, and connect the circled pivot value in one step with its X location in the next). At the bottom, copy down the last value in the column to show the sorted array.

Initial array:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 10 | 9 | 1 | 3 | 5 | 8 | 11 | 15 | 7 | 4 | 14 | 2 | 13 | 12 | 6 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | 4 | 1 | 3 | 5 | 6 | 11 | 15 | 7 | 9 | 14 | 10 | 13 | 12 | 8 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | 4 | 1 | 3 | 5 | 6 | 7 | 8 | 11 | 9 | 14 | 10 | 13 | 12 | 15 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 2 | 1 | 3 | 4 | 5 | 6 | 7 | 8 | 11 | 9 | 14 | 10 | 13 | 12 | 15 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 11 | 9 | 10 | 12 | 13 | 14 | 15 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |

Final array:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |