PL UID: dev

Question 1: heap_shorts_array_implementation_mc

{"heap_array": [{"key": "a", "html": "\n The third-smallest element is at either ${\\tt arr[2]}$ or ${\\tt arr[3]}$.\n "}, {"key": "b", "html": "\n Using the heap to sort the array is asymptotically as efficient as using mergeSort.\n "}, {"key": "c", "html": "\n Swapping the last and the second last element in the array results in another valid heap.\n "}, {"key": "d", "html": "\n The heap order property may be violated by swapping the contents of two adjacent locations in ${\\tt arr}$ so that\n the first is now smaller than the second.\n "}]}

Question 1.1: heap_array

INCORRECT: 0

[{'key': 'b', 'html': '\n Using the heap to sort the array is asymptotically as efficient as using mergeSort.\n '}, {'key': 'd', 'html': '\n The heap order property may be violated by swapping the contents of two adjacent locations in ${\\tt arr}$ so that\n the first is now smaller than the second.\n '}]

['a', 'c']

Question 2: heap_shorts_build_heap

{"input": {"_type": "dataframe", "_value": {"data": [[85, 32, 88, 61, 52, 5, 46, 37, 13, 72]], "index": ["key"], "columns": [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]}}}

Question 2.1: arr

INCORRECT: 0

[5, 13, 46, 32, 52, 88, 85, 37, 61, 72]

[89, 8, 890, 89, 890, 809, 809, 89, 89, 89]

Question 3: heap_shorts_operations

{"graph": "strict digraph {\n\tnode [label=\"\\N\"];\n\t0\t [label=16];\n\t1\t [label=21];\n\t0 -> 1;\n\t2\t [label=20];\n\t0 -> 2;\n\t3\t [label=54];\n\t1 -> 3;\n\t4\t [label=53];\n\t1 -> 4;\n\t5\t [label=23];\n\t2 -> 5;\n\t6\t [label=92];\n\t2 -> 6;\n\t7\t [label=89];\n\t3 -> 7;\n\t8\t [label=95];\n\t3 -> 8;\n\t9\t [label=83];\n\t4 -> 9;\n\t10\t [label=59];\n\t4 -> 10;\n}\n", "removed_values": "<pl-answer correct='true'>16</pl-answer>\n<pl-answer correct='true'>20</pl-answer>\n<pl-answer>21</pl-answer>\n<pl-answer correct='true'>23</pl-answer>\n<pl-answer>53</pl-answer>\n<pl-answer>54</pl-answer>\n<pl-answer correct='true'>59</pl-answer>\n<pl-answer>83</pl-answer>\n<pl-answer>89</pl-answer>\n<pl-answer>92</pl-answer>\n<pl-answer>95</pl-answer>", "inserted_values": "<pl-answer>16</pl-answer>\n<pl-answer>20</pl-answer>\n<pl-answer correct='true'>21</pl-answer>\n<pl-answer>23</pl-answer>\n<pl-answer correct='true'>53</pl-answer>\n<pl-answer>54</pl-answer>\n<pl-answer correct='true'>59</pl-answer>\n<pl-answer>83</pl-answer>\n<pl-answer>89</pl-answer>\n<pl-answer>92</pl-answer>\n<pl-answer>95</pl-answer>", "heap_manipulation_insert": [{"key": "a", "html": "16"}, {"key": "b", "html": "20"}, {"key": "c", "html": "21"}, {"key": "d", "html": "23"}, {"key": "e", "html": "53"}, {"key": "f", "html": "54"}, {"key": "g", "html": "59"}, {"key": "h", "html": "83"}, {"key": "i", "html": "89"}, {"key": "j", "html": "92"}, {"key": "k", "html": "95"}], "heap_manipulation_remove": [{"key": "a", "html": "16"}, {"key": "b", "html": "20"}, {"key": "c", "html": "21"}, {"key": "d", "html": "23"}, {"key": "e", "html": "53"}, {"key": "f", "html": "54"}, {"key": "g", "html": "59"}, {"key": "h", "html": "83"}, {"key": "i", "html": "89"}, {"key": "j", "html": "92"},

{"key": "k", "html": "95"}]}

Question 3.1: heap_manipulation_insert

INCORRECT: 0

Expected: [{'key': 'c', 'html': '21'}, {'key': 'e', 'html': '53'}, {'key': 'g', 'html': '59'}]

f

Question 3.2: heap_manipulation_remove

INCORRECT: 0

Expected: [{'key': 'a', 'html': '16'}, {'key': 'b', 'html': '20'}, {'key': 'd', 'html': '23'}, {'key': 'g', 'html': '59'}]

b

Question 1: disjt_unions

{"n": 15, "arr": {"_type": "dataframe", "_value": {"data": [[3, -1, 14, -11, 14, 14, 7, -3, 3, 7, 3, 12, 3, 12, 3]], "index": ["arr"], "columns": [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14]}}, "bigdata": 82309, "bigunions": 44123}

Question 1.1: height

INCORRECT: 0

Expected: 2

890

Question 1.2: unions

INCORRECT: 0

Expected: 12

908

Question 1.3: uptreenum

INCORRECT: 0

Expected: 38186

890

Question 1.4: files

INCORRECT: 0

mdtest.md

Phrase Emphasis

Markdown: Some of these words are emphasized. Some of these words are emphasized also. Use two asterisks for strong emphasis. Or, if you prefer, use two underscores instead.

Lists

- Red
- Green
- Blue

Ordered (numbered) lists use regular numbers, followed by periods, as list markers:

1. Red
2. Green
3. Blue

Code

```
This is a code indent!
```

To specify an entire block of pre formatted code, indent every line of the block by 4 spaces or 1 tab. Just like with code spans, &, <, and > characters will be escaped automatically.

This is a blank page.

picture.png

This is a blank page.

playListAns.cpp

```
void playList(vector<char> &A)

{

    bool Tay = // YOUR CODE HERE!!

        for (int i = 1; i < A.size(); i++)

    {

        int next = findNext(A, i, Tay);

        swap(A[next], A[i]);

        Tay = !Tay;

    }

}
```

This is a blank page.