SwitchAnimation component requires a specific data structure, trace, to make it run.

A trace is an array that contains steps, a step is an array of bars, and a bar is an JavaScript object which contains 7 properties to to present its height, colour and relative position on the screen.

A trace example shows below. In this example trace, there are 1 step with 2 bars in it. Each bar contains value，height， backgroundColor，isPivot， y，x and key.

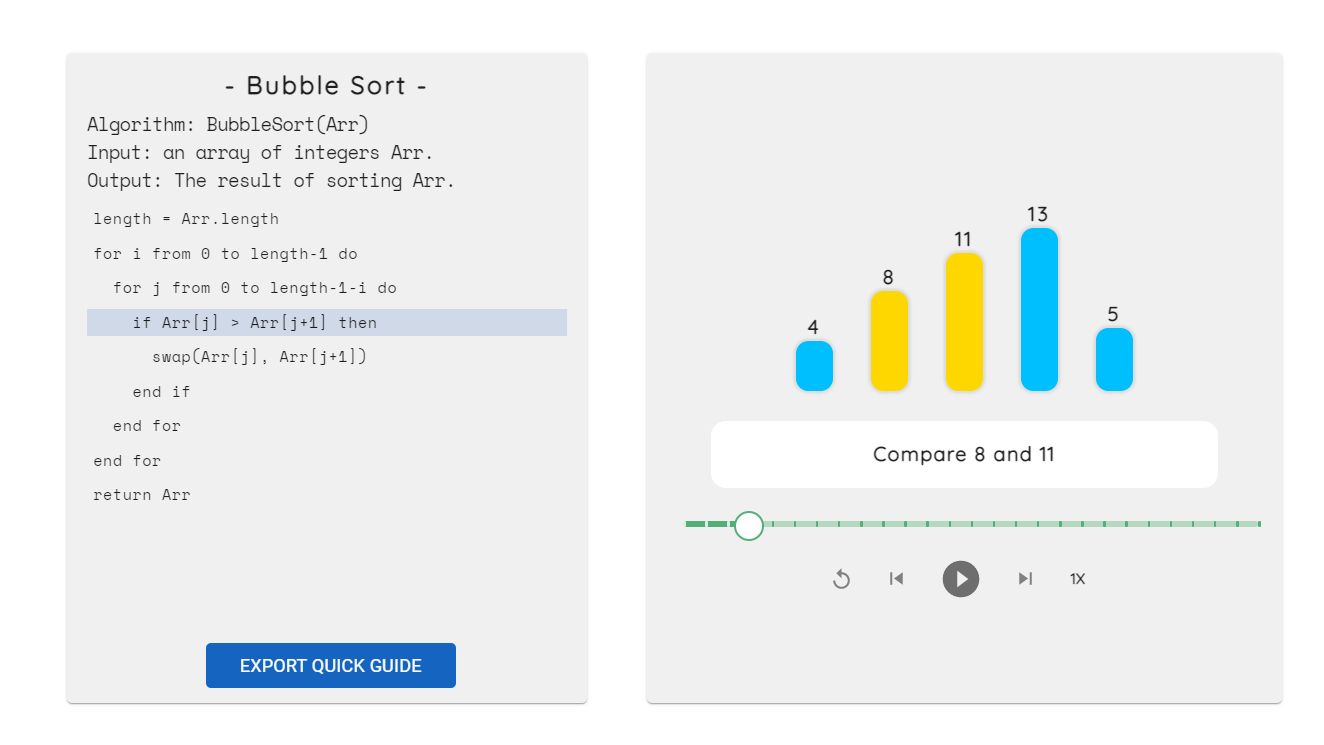
const trace = [  
  // trace  
 [  
    // step  
 {   
      // bar  
      value: 1, height: 10, backgroundColor: COLORS.original, key: 0, y: 0, isPivot: false, x: 0   
   },  
   {   
      // bar  
      value: 2, height: 20, backgroundColor: COLORS.original, key: 1, y: 0, isPivot: false, x: 0   
   },  
 ],  
];

Function patch can convert an integer array to a trace with 1 step. The integer array will be generated either randomly or through InputBar component from the user.

Then, apply the corresponding sorting algorithm to generate the following steps.

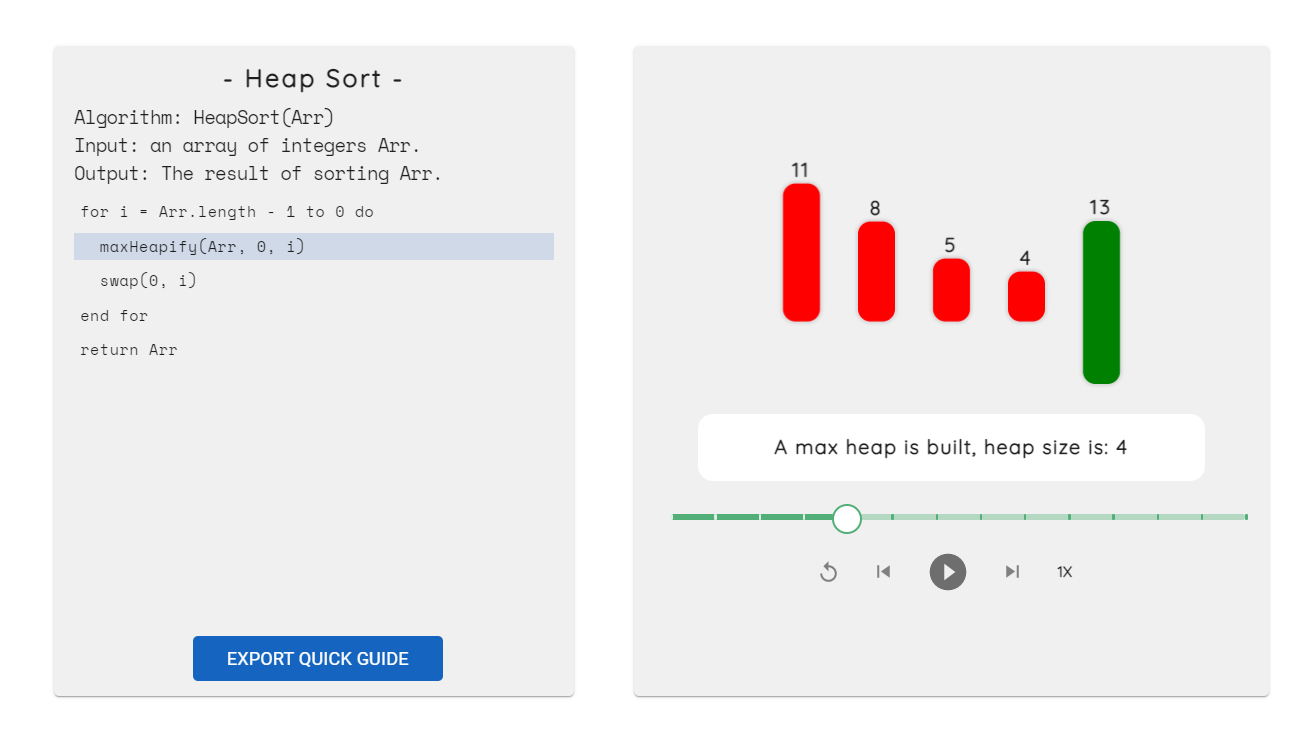
Generally used functions and variables, for example, COLORS, changeColor, hardcopy are packed to patch.jsx to reduce coupling. COLORS is a dictionary to store colours, changeColor will change the bar’s colour, and changeY will change a bar’s ordinate.

I Can Sort supports 6 types of sorting algorithms. Bubble sort and selection sort are mainly implemented with swap and changeColor function.



Screenshots of ’Bubble sort’ animation

Implementations of heap sort and insertion sort used changeY function to perform a more complicated animation.



Screenshots of ’Heap sort’ animation

An example of the use of changeY function in heap sort shows below.

 // build max heap

for (var j = 0; j < i + 1; j++) {

    changeColor(patched, j, COLORS.current);

    changeY(patched, j, -50);

}

Implementations of merge sort and quick sort are implemented using recursion. An example of the use of recursion in quick sort shows below.

 // Sorting...

recursiveQuickSort(patched, 0, arr.length - 1, description, trace);