

Team Chef
Kenny Chen
Jasper Cheung
Grace Cuenca
p.d.l.
UMM

Visualizer

```
+ int startYUI;  
+ int startYAlgo;  
+ ArrayList<Button> structures;  
+ ArrayList<TextBox> textBoxes;  
+ TextBox focus;  
+ ArrayList<Button> userInput;  
+ ArrayList<Button> algorithms;  
+ ArrayList<ArrayList<Integer>> steps;  
+ int step;  
+ ArrayList<Button> stepsUI;  
+ int dataStructure;  
+ ArrayList<Integer> a1;  
+ LinkedList<Integer> s11;  
+ LinkedList<Integer> d11;  
+ Stack<Integer> st;  
+ BST bt;  
+ ArrayDeque<Integer> que;  
+ ArrayDeque<Integer> deque;  
+ ALHeap minHp;  
+ void setUp();  
+ void draw();  
+ void mousePressed();  
+ void keyPressed();  
+ void changeStructure(int actionID);  
+ void inputAction(int actionID);  
- Integer numFromTextBox(int numTextBox);  
+ void algorithmAction(int actionID);  
+ void deepCopyStep();  
+ void arrowAction(int actionID);  
+ void bubbleSort();  
+ void selectionSort();  
+ void insertionSort();
```

ArrayList

```
+ void displayAl();  
+ void displayAlUI();
```

BinarySearch Tree

```
+ void displayBTL();  
+ void displayBTreeNode  
  (TreeNode tn, int layer,  
   int i, float y, int size);  
+ float calculateX  
  (int layer, int i);  
+ void displayBTUI();
```

Button

```
+ int x, y, len, wid;  
+ String text;  
+ int actionID;  
+ Button(int initX,  
  int initY, int initLen,  
  int initWidth,  
  String initText,  
  int initAction);  
+ void display;  
+ boolean contains  
  (int pointX, int pointY);
```

Deque

+ void displayDeque();

MinHeap

+ void displayMinHeap();
+ void displayMinHeap
(int index, int layer, int i,
float y, int size);

Stack

+ void displaySt();
+ void displayStUI();

LL

+ void displaySll();
+ void displaySllUI();
+ void displayDll();
+ void displayDllUI();
+ void drawArrow
(int x1, int y1, int x2, int y2);

Queue

+ void displayQueue();
+ void displayQueueUI();

Text Box

+ int x, y, len, wid;
+ String label;
+ String text = " ";
+ TextBox(int initX, int initY,
int initLen, int initWid,
String initLabel);
+ void display();
+ boolean Contains(int pointX,
int pointY);

ALHeap

```
+ ArrayList<Integer> _heap;  
+ ALHeap();  
+ String toString();  
+ boolean isEmpty(); // heap size  
+ Integer peekMin();  
+ void add(Integer addVal);  
+ Integer removeMin();  
- int minChildPos(int pos);  
- Integer minOf(Integer a, Integer b);  
- void swap(int pos1, int pos2)
```

BST

```
+ TreeNode root;  
+ BST();  
+ TreeNode getRoot();  
+ void insert(int newVal);  
+ void preOrderTrav();  
+ void inOrderTrav();  
+ void postOrderTrav();  
+ TreeNode search(int target);  
+ TreeNode search(int target, TreeNode currNode);  
+ int height();  
+ int height(TreeNode currNode);  
+ int numLeaves();  
+ int numLeaves(TreeNode currNode);  
+ TreeNode remove(int remVal);  
+ boolean isLeaf(TreeNode node);
```

Tree Node

```
+ int - cargo;  
+ TreeNode - lt, - rt;  
+ TreeNode(int initValue);  
+ TreeNode(int initValue, TreeNode initLeft,  
    TreeNode initRight);  
+ TreeNode getLeft();  
+ TreeNode getRight();  
+ int getValue();  
+ void setLeft(TreeNode theNewLeft);  
+ void setRight(TreeNode theNewRight);  
+ void setValue(int theNewValue);
```