Using the Binary Search Algorithm and trace table examples at the end of this document, complete the trace tables below for the Color Array. Upload this document to github and submit the link to your repository to the dropbox.

1st search: violet

|  |  |  |  |
| --- | --- | --- | --- |
| **First** | **Last** | **Middle** | **Comparison** |
| 0 | 10 | 5 | Violet>indigo |
| 6 | 10 | 8 | Violet>red |
| 8 | 10 | 9 | Violet=Violet **Return: true** |
|  |  |  |  |
|  |  |  |  |

Set first to 0

Set last to length-1

Set found to FALSE

While (first <= last AND NOT found)

Set middle to (first + last)/2

IF (item equals data[middle]))

Set found to TRUE

ELSE

IF (item < data[middle])

Set last to middle -1

ELSE

Set first to middle +1

RETURN found

2nd search: green

|  |  |  |  |
| --- | --- | --- | --- |
| **First** | **Last** | **Middle** | **Comparison** |
| 0 | 10 | 5 | Green<Indigo |
| 0 | 5 | 2 | Green >chartreuse |
| 2 | 5 | 3 | Green > Dark brown |
| 3 | 5 | 4 | Green=Green **Return: true** |
|  |  |  |  |

Set first to 0

Set last to length-1

Set found to FALSE

While (first <= last AND NOT found)

Set middle to (first + last)/2

IF (item equals data[middle]))

Set found to TRUE

ELSE

IF (item < data[middle])

Set last to middle -1

ELSE

Set first to middle +1

RETURN found

3rd search: yellow

|  |  |  |  |
| --- | --- | --- | --- |
| **First** | **Last** | **Middle** | **Comparison** |
| 0 | 10 | 5 | Yellow>indigo |
| 6 | 10 | 8 | Yellow>red |
| 8 | 10 | 9 | Yellow>violet |
| 9 | 10 | 10 | Yellow=Yellow **Return: true** |
|  |  |  |  |

Set first to 0

Set last to length-1

Set found to FALSE

While (first <= last AND NOT found)

Set middle to (first + last)/2

IF (item equals data[middle]))

Set found to TRUE

ELSE

IF (item < data[middle])

Set last to middle -1

ELSE

Set first to middle +1

RETURN found

**Color array**:

|  |  |
| --- | --- |
| aqua | [0] |
| brown | [1] |
| chartreuse | [2] |
| dark brown | [3] |
| green | [4] |
| indigo | [5] |
| lavender | [6] |
| magenta | [7] |
| red | [8] |
| violet | [9] |
| yellow | [10] |



Above: Binary Search Algorithm

