Ramsey theory and Rado’s numbers

Adrian Galicia

Mentor: Kellen Myers
Objective

- Create an algorithm to compute Rado numbers
- Implement the algorithm in a web-friendly language (javascript, PHP, etc.)
- Visualizing the computation
- Prove the output (Rado numbers) with certificates.
Given some edge-coloring of a complete graph, Ramsey’s Theorem states that there exist monochromatic sub graphs (depending on how many colors are used).

If two colors are used (red & blue), for any positive integers \((r,s)\), there exist a positive integer \(N=R(r,s)\) where a coloring of \(K[N]\) will give \(K[r]\) red or \(K[s]\) blue.

The \(R(r,s)\) is smallest integer \(N\) for which the theorem holds.
Ramsey’s Theorem example
Ramsey’s theorem example
Given some equation “E” such as $x+y = z$ and some number of colors “r” ($r=2$, red and blue in this case), find the $N$ so that there is no way to color 1 through $N$ without a solution to the equation $E$ of the same color.

This is analogous to $N=R(r,s)$, like $R(3,3)=6$. As showed in the graph, $K[5]$ is the smallest number you can color and avoid monochromatic $K[3]$. 
Rado number example

N = 1

N = 2

N = 3

N = 4

N = 5
Tree output

\[ x + y = z \]

Can be omitted. mirror image.
Ultimate goal

- Make a program that gives out the least Rado number given some equation taken from the user input.
- If possible, it will not only work with 2 colors but with 3 or more colors.
- It will have a tree stylized output (certificate).
Languages used

- HTML
- CSS3
- JavaScript
Demonstration

- rado_number_1.html
- Main Functions
  1. calculateN()
  2. checkColoring()
  3. intoColors()
  4. drawTree() (not finished)
If valid a zero is added (red number) else the last color gets replaced by the other color
### How does it work (2)

#### Coloring: 1, 2, 3, 4, 5, 6

<table>
<thead>
<tr>
<th>2(n)</th>
<th>X</th>
<th>Y</th>
<th>X+Y combs.</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>2(n)</td>
<td>4(n)</td>
<td></td>
<td></td>
<td>7(n)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>8, 10</td>
<td>8, 16, 20</td>
<td>12, 20, 24, 16, 24, 28, 14, 28, 32</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>6, 12</td>
<td>4, 12, 24</td>
<td>6, 14, 26, 10, 18, 30, 7, 21, 42</td>
<td></td>
</tr>
</tbody>
</table>

If a number in the 3rd column is in the 4th column, then that is the monochromatic solution to the equation which means that the coloring is not valid.

\[2x + 4y = 7z\]
intoColors()

- Takes the string of zeros and ones \([0,1,1,0,0]\) and converts them into a numbers with colors \([1, 2, 3, 4, 5]\) to be outputted in the screen.
- If a coloring is not valid it highlights the values for which the coloring has a monochromatic solution for a given equation.

\[
1 + 2 = 3, \quad 4 + 5 = 5
\]
drawTree()

- Takes the biggest non extensible colorings and orders them by length.
- Makes a list structure of the longest one.
- Then compares it to the next coloring and at the location where they differ adds a colored number.
- And then a CSS style sheet changes its appearance to an actual tree.
Another tree

\[ x + 2y = z \]
Future Work

- Finish the tree output
- Add feature of testing a single coloring on demand.
- Test several equations
- More variables