```
<!DOCTYPE html>
<html>
<head>
<title>EECS 492 A1 Results</title>
</head>
<body>
<p>
EECS 492 A1 Results<br>
Mickey Chao<br>
September 29, 2015<br>
</p>
<h2>A1 Results</h2>
<p>
The purpose of this project is to recreate
<a href="http://rogeralsing.com/2008/12/07/genetic-programming-
evolution-of-mona-lisa/">Roger
Alsing's work on genetic programming to create art</a>.
The population consists of approximations to a given image.
Each image uses P polygons, where P=100 for the results on this
page.
N is the population size and K is the number of new children
created per generation.
T is the number of generations displayed in each animation.
The fitness of the image is a measure of how closely it matches
the original image.
The learning curve shows how the fitness of the best
approximation improves over time.
The 3D graphs depict how the fitness after 25000 generations
varies with N and K.
</p>
<table border=1>
<tr>
<th>Original image</th>
<th>Approximation</th>
<th>Learning curve</th>
</tr>

<tr>
<td>216&times;174 pixels<br><img src="darwin.bmp"></td>
<td>P=100, N=1, K=1<br><img src="darwin/darwin_anim.png"></td>
<td><img src="darwin_fitness.png"></td>
<td><img width=400 height=400 src="darwin_tuning.png"><br>N=8,
K=8 gave the best result<td>
</tr>
```

```
<tr>
<td>128&amp;times;128 pixels<br><img src="haystack.jpg"></td>
<td>P=100, N=1, K=1<br><img
src="haystack/haystack_anim.png"></td>
<td><img src="haystack_fitness.png"></td>
<td><img width=400 height=400 src="haystack_tuning.png"><br>N=2,
K=8 gave the best result<td>
</tr>

<tr>
<td>328&amp;times;189 pixels<br><img src="michigan.bmp"></td>
<td>P=100, N=1, K=1<br><img
src="michigan/michigan_anim.png"></td>
<td><img src="michigan_fitness.png"></td>
<td><img width=400 height=400 src="michigan_tuning.png"><br>N=4,
K=1 gave the best result</td>
</tr>

<tr>
<td>32&amp;times;32 pixels<br><img src="mona_lisa.bmp"></td>
<td>P=100, N=1, K=1<br><img
src="mona_lisa/mona_lisa_anim.png"></td>
<td><img src="mona_lisa_fitness.png"></td>
<td><img width=400 height=400 src="mona_lisa_tuning.png"><br>N=2,
K=2 gave the best result</td>
</tr>

<tr>
<td>128&amp;times;128 pixels<br><img src="turing.jpg"></td>
<td>P=100, N=1, K=1<br><img src="turing/turing_anim.png"></td>
<td><img src="turing_fitness.png"></td>
<td><img width=400 height=400 src="turing_tuning.png"><br>N=4,
K=2 gave the best result</td>
</tr>

</table>
<img width=600 height=600 src="overall3d.svg"><br>
N=1, K=2 gave the best result overall after 25000 generations<br>
<hr>
<p>
Mutations used:
<ul>
<li>With 1/5 probability, replace a polygon with a random
```

triangle.
With 1/5 probability, swap the order of two polygons which are in adjacent depth layers.
With 3/5 probability, mutate an existing polygon:

With 1/2 probability, alter the color slightly.
With 1/2 probability, alter the shape of the polygon:

With 1/3 probability, remove a vertex.
With 1/3 probability, add a vertex at a random point.
With 1/3 probability, move a vertex slightly.

Coordinates of vertices were moved using a Gaussian distribution of mean 0 and standard deviation IMG_SIZE/5 pixels.
Components of colors (0-255 scale) were altered using a Gaussian distribution of mean 0 and standard deviation 25.

Random triangles were created by picking a point on the image with uniform distribution,
then adding a Gaussian distribution of mean 0 and
standard deviation IMG_SIZE/10 of the image width to the point to obtain the vertices.
Random colors were picked using uniform distribution over the RGB color space.

Parents were selected from an array of individuals sorted in decreasing order of fitness
using the index <code>Rand.nextInt(Rand.nextInt(n) + 1)</code> where Rand.nextInt(m) returns a
random integer uniformly distributed between 0 and m-1 inclusive.
This made the more fit individuals
more likely to be chosen for crossover.

</body>
</html>