babaroga LLC.

Thank you for your interest in Babaroga's software engineering test! At Babaroga we like to hire developers that are passionate about what they do. Programming is a skill that can be learned – so we look for applicants that show creative problem solving skills, are humble (yet stand firm when they believe in something), and are fun to work with.

Babaroga hires programmers with varying degrees of experience, so whether you are a fresh noob straight out of university or a seasoned veteran of several hundred software battles, we are interested in seeing what you can do. If you feel you aren't experienced enough to answer a question, just try to answer it with your finely-tuned creative reasoning skills…

The test is designed to get progressively harder. Please write out all the steps you take to achieve your answers. Also, be prepared to discuss your answers if you are awesome enough to come in for an interview.

Please return the results as soon as available to our team by emailing to one of the following email addresses:

Chicago, IL Applicants: contact@babaroga.com
London, UK Applicants: dbachowski@babaroga.com

Please include all source code as well as compiled binaries if applicable.

Sincerely,
Babaroga

babaroga LLC.

1) Warm-ups

How many bits are there in a 32-bit integer?

How many bytes are there in 1 gigabyte?

If a woodchuck chucks 2 woods a day, but improves his chucking ability by 50% every 2 days, how many days would it take him to chuck 1000 woods? This is assuming a woodchuck *can* in fact chuck wood.

2) Object-Oriented Design

Design an object model for a chicken. What kinds of members and methods should it have? Now imagine your program must simulate an entire farm. Give an example of a few classes (5-6) you would use, and describe how they would relate to one another. You do not need to describe the new classes' methods or members, nor do you need to describe the entire farm in detail.

3) 2D Math:

Given the following structs, complete the Rotate() function below in C#. You may use any basic .Net libraries you wish.

```
struct Rectangle
{
	/* Each point represents one of the Rectangle's corners. */
	Point p1, p2, p3, p4;

	void Rotate(Point origin, float angle)
	{
		/* fill in */
	}
}

struct Point
{
	float x, y;
}
```

Chicago, IL
312·375·1946 • contact@babaroga.com • www.babaroga.com

4) Image Cropper

The following application should be written in C/C++. Do not use external image libraries.

Write a commandline tool that crops an image to a smallest bounding rectangle. Smallest bounding rectangle is defined by edges of the image that are touching the unique pixels of the image. You can observe smallest bounding rectangle in **fozzie-boundingrect.bmp**. You can assume that the image is an 8-bit image.

fozzie-in.bmp is a sample 8-bit image containing large amount of magenta, which is considered to be transparent. The tool should take **fozzie-in.bmp**, and output **fozzie-out.bmp**, which is the image with the smallest bounding rectangle. These images are included in your .zip test package.

fozzie-boundingrect.bmp

fozzie-in.bmp

fozzie-out.bmp