# Snapper Coding Challenge

It's April 1, 2242. Your job is to save the world.

Well, a little world. Specifically, the asteroid Z8483-X-32 that you and Alphonso Snappageri are stuck on. You've been stranded there ever since the evil Vlados hit your spaceship with a nuclear torpedo fired from one of their spaceships. Now you and Alphonso are trying to save your little world from a concerted Vlados attack.

The main problem you have is detecting the Vlados spaceships and nuclear torpedos, because they're protected with cloaking devices. Alphonso has invented an imaging anti-neutrino system (which he has modestly named the "Snapper") that provides the only information you have about their location, but it's not very good information.

First, the Snapper only detects whether there are anti-neutrinos at any particular point on an image, not what their intensity is. In other words, the data it provides is the equivalent of a black-and-white image.

Second, the data is very noisy – even if there are no targets in a particular area, some pixels will be "on", and if there is a target, some of its pixels will be "off". For example, here's a 20 x 20 sample of raw data from the Snapper (where each "+" is a pixel that is on):
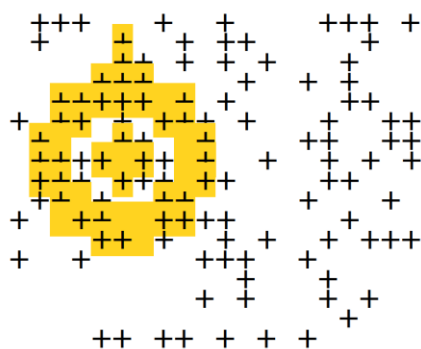
Below is a sample image of a nuclear torpedo:
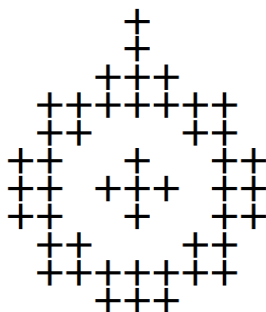
On the Snapper data, we've highlighted the pixels that should be "on" for a nuclear torpedo.

You can see that more of the highlighted pixels are "on" in the highlighted area than in other areas of the image. You must use this difference to locate the targets in the Snapper data. Along with this document you've received three files, all text files using "+" symbols to represent "on" pixels and spaces to represent "off" pixels:

1. TestData.blf: a 100 x 100 swath of raw Snapper data containing between four and ten targets.
2. NuclearTorpedo.blf: a perfect image of a nuclear torpedo.
3. Starship.blf: a perfect image of a Vlados starship.

Your task is to do the following:

1. Design and write a C# program that can analyze arbitrary-sized Snapper images, returning a list of targets found. Each target found should include the target type found (starship or nuclear torpedo), the coordinates of the target on the Snapper data, and some indication of your confidence in the target detection.

2. Design and write test code that submits the test data to your package and prints the results returned by your program.

3. After defeating the Vlados using your target data, submit your program and the test results back to me by email at bruno.martins@mottmac.com. If your code fails to detect the targets, submit it anyway – we'd like to see how you attempted it.