Code Test: Space Defense

About This Test

The purpose of this code test is to show us your skills in:

- Knowledge of OOP concepts, and judicious use of them
- Algorithms and problem-solving
- Code structure and commenting

Please keep these aspects in mind as you develop your solution. Also, your chosen algorithm doesn't necessarily have to be the best you can think of, but one that you can implement in the allocated time.

Instructions

This test should be completed in under 2 (two) hours, and should be in PHP unless previously agreed otherwise. All classes, methods and so on should be commented in PHPDoc format, and you should include a generous comment block explaining your algorithm for part 2 before the bulk of this code.

Background

You are the admiral of a mighty space fleet comprised of 50 vessels. Your fleet consists of two major types of vessels - support craft and offensive craft. Vessels can all receive a command that tells them to move to given co-ordinates.

There are three different types of support craft - refueling, mechanical assistance and cargo. They all carry a medical unit. Each vessel can receive orders related to each of the tasks it can carry out.

There are also three different types of offensive craft - battleships, cruisers and destroyers. Battleships have 24 cannons, destroyers have 12 and cruisers have 6. Each offensive craft can receive an attack command, which will fire all its cannons. They can also be instructed to raise their shields.

Finally, the fleet has a command ship, which is where you are. The command ship is one of the battleships, and there is only one per fleet.

Part 1

(Recommended time: 45 minutes)

Define a set of data structures to accurately reflect this fleet. Make sure that new types of vessels can be added to your fleet with minimal effort.

Part 2

(Recommended time: 1 hour and 15 minutes)

You are taking your fleet, made up of an equal number of offensive and support ships, to your assigned deployment point when you are ambushed by enemy forces. Your defense tactic is to pair each support ship with one offensive ship in order to share the offensive ship's shield.

Assuming a two-dimensional layout with a maximum size of 100x100, write some code that is able to represent your fleet location data and populate it with your 50 ships in random positions. Then, implement an algorithm that generates 25 pairs of ships, and issues the commands to make the pairs occupy adjacent positions on the grid by moving one or both ships. Your vessels need to assume this defensive formation as quickly as possible, so you will need to find an algorithm that gives an optimized set of pairs, but that is also quick to generate them.

Questions

Please also supply your answer to the following questions:

1. What is the complexity of your algorithm (in big O notation)?
2. How would you improve your algorithm?
3. How would your adapt your algorithm to three dimensions? How would that affect the complexity?