Battleship Bots

Problem Solving II

Q) What is the problem?

Q) What is the problem?

A) My ship is still getting sunk!

Q) What is the problem?

A) My ship is still getting sunk!

Can we be a bit more precise?

What is that you would like your ship to do?

Maybe behave more like the **Pack**…

What does the **Pack** do?

Maybe behave more like the **Pack** …

What does the **Pack** do?

Watch the **Pack** for 5 minutes and then write down what it is doing on the Whiteboard…

So, what is the **Pack** doing?

So, what is the **Pack** doing?

Pack

- Moves towards its friends
- Fires at the nearest enemy ship

So we set this as our goal

- Move towards the friend ships
- Fire at the nearest enemy ship

So we set this as our goal

- Move towards the friend ships
- Fire at the nearest enemy ship

In Software Engineering we refer to these as our requirements and we place them in a numbered list

Requirements

Number	Description	Priority
1	Move towards the friend ships	Medium
2	Fire at the nearest enemy ship	High

Note that we have added a **priority** to the requirements. It is more important that we fire at the nearest enemy that move towards them as hitting the nearest enemy gains you marks

Once we have our requirements how are we going to know if we have implemented them correctly?

Once we have our requirements how are we going to know if we have implemented them correctly?

We need to consider how we are going to test the code that we produce…

Each requirement requires at least one test associated with it to prove that the requirement has been implemented correctly

Each requirement requires at least one test associated with it to prove that the requirement has been implemented correctly

We call these tests the **Acceptance Tests**

Test 1	Move towards the friend ships
Pre-requisite	Battleship Bots server running
Requirements covered	1

Step	Action	Outcome
1	Start your Battleship	Your Battleship should move towards its friends

Test 2	Fire at the nearest enemy ship
Pre-requisite	Battleship Bots server running
Requirements covered	2

Step	Action	Outcome
1	Start your Battleship	
2	Wait 10 minutes	The nearest enemy should be getting hit

In Software Engineering we refer to these tests as **Acceptance Tests**

They are always **agreed with the customer** at the beginning of a project

They are used to **prove** that the required software has been delivered

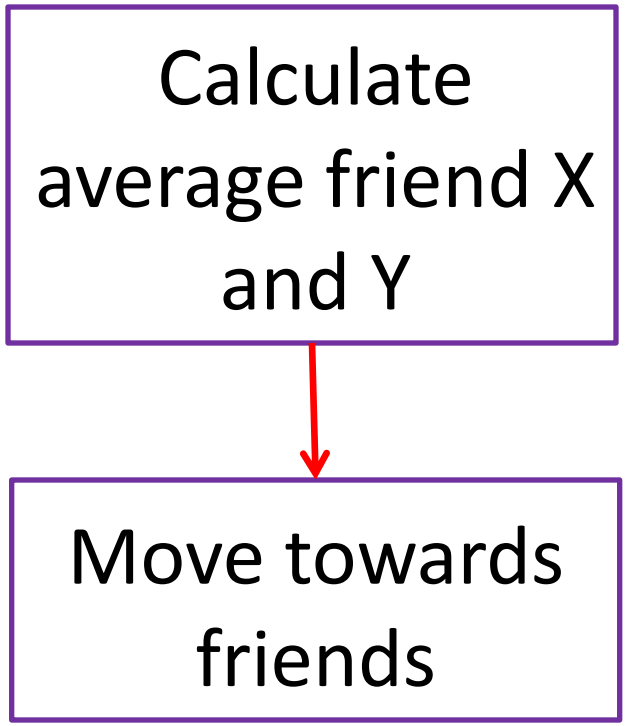
They trigger **payment**

Next we need to think about how we are going to implement the requirements

Remember them…

1. Move towards the friend ships
2. Fire at the nearest enemy ship

In both requirements we need to know where the friend ship are…



In Software Engineering we refer to this process as **Design**

Next we need to think about how to carry out **Implementation**

In tactics()…

You will need to call two routines

- calculate_average_friend_X()
- calculate_average_friend_Y()

These routines will need to be written and placed above the tactics() routine

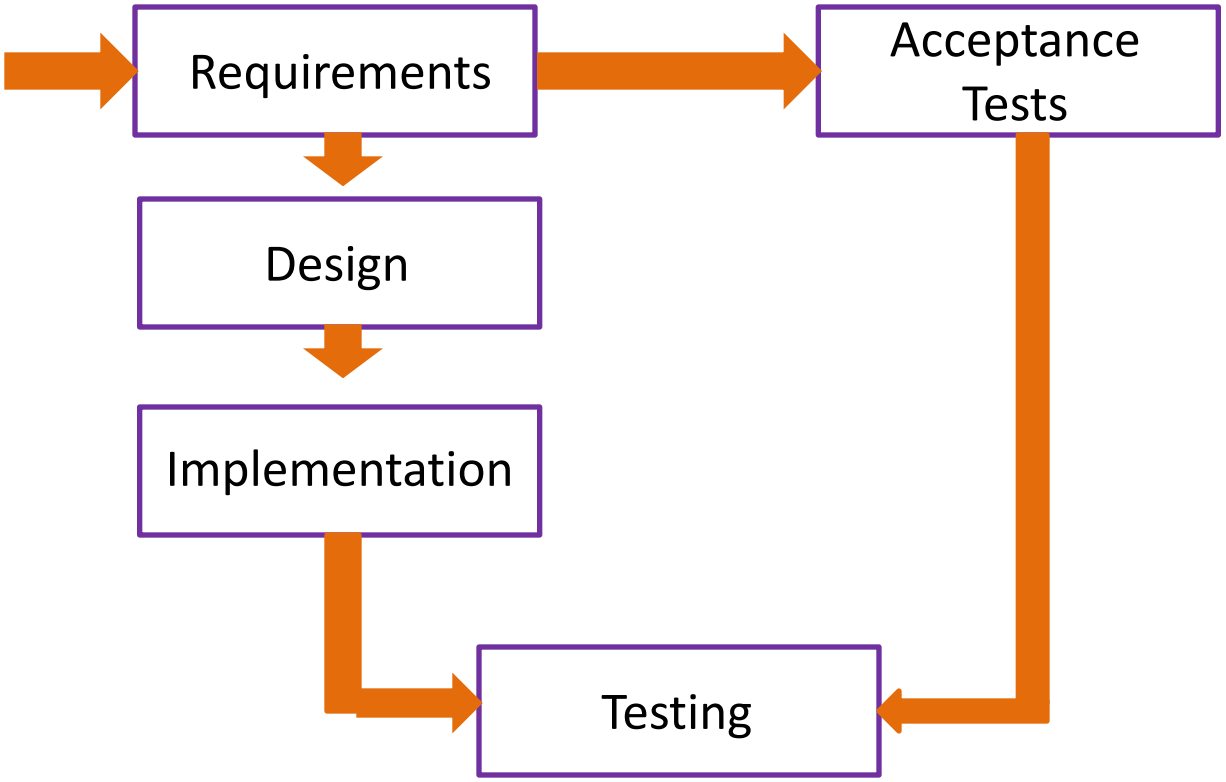
When these routines are written and placed in the correct place the **Acceptance Tests** can be ran

You will need friends (allies)

Choose three people that you trust

No group should number more than four

This process used to solve the Battleship problem can be viewed as follows…



If you need help calculating average ship X and Y…

- Ask your **tutor** for help
- Ask your **PAL leaders** for help
- Ask **EspressoProgramming** for help

What else should you do?

The End