Test data | Part A

When creating a system it's really important to test that it does what it is supposed to do. It is important to test if the system fails, it does so cleanly when presented with incorrect input. With a good design and carefully coded algorithms, most errors should be avoided but testing is essential to check that everything works as it should. If the system is not thoroughly tested and bugs remain in it after you have released it, you may end up with some very confused users, or much worse.

There are many different methods of testing, and these methods should be used throughout the development of a system, as well as at the end of development. For each test you do, you should state why you are doing the test, and what you expect the outcome to be.

Let's pretend I have coded a system that works out how much I should pay an employee for the day. I have to type in how many hours they've worked that day, and the system tells me a money amount that I should pay them. In this scenario, let's pretend I pay my employees £5 per hour. Let's go through some different test cases. For a valid test I should use a value that I would expect somebody to type into my system. For example, I might type in an 8, because on average most employees will work 8 hours in a day. My expected result is that the system should output the value of £40 as 5 x 8 is 40.

Test data | Part A