# Betting Odds Tester

Create an application in any suitable modern language (e.g. C#, JavaScript, TypeScript) that implements the following specification:

A gaming arcade wishes to implement a new betting machine and requires a software simulation to test whether candidate betting machines provide a suitable level of profit. The betting machine is not allowed to exceed 10% profit on average and should not make a loss.

The user will launch the program to test a particular betting machine and will select the number of random tests to run on that machine. The application should output the overall profit/loss and percentages from these tests.

e.g.: 10,000 tests were run; Profit: £874.25; Profit Percentage: 8.4%

The two candidate betting machines currently being considered are:

Black Or Red Party

This is a roulette-style machine, consisting of a disk with 62 evenly-spaced slots. 30 are red, 30 are black and 2 are white. A spinning ball stops randomly in one of these slots: each slot has an equal chance. The rules are these:

* The stake is £1
* The user selects red or black
* The user gets back double the stake (£2) if the ball stops in a slot of their selected colour
* The user gets back 10% of their stake (£0.10 if the ball lands on a white.

Dice Roll 2000

This is a simple dice roll simulation machine consisting of two six-sided dice being rolled once. The following rules apply:

* The stake is £1
* The user only wins if a double is thrown
* The user wins one and a half times the pound face value of one of the winning dice – e.g. if two sixes are thrown, the user wins £6\*1.5=£9, whereas if two ones are thrown, the user wins £1\*1.5=£1.50.

Which of these two betting machines would be most appropriate to implement? How many tests do you recommend the users run to get sufficient confidence in the results?

Please note the following points:

* The application must be extensible enough to allow the addition of extra betting machine candidates to be as easy as possible.
* The application should be suitably testable.
* The user of this application will be a power user: a friendly UI is not a requirement.

The complete solution source code should be sent to PXtech in good time prior to your interview, during which you will walk the interviewers through your solution.