

.NET Developer Test

Simplified Lottery Game

This is a short task that aims to evaluate your problem-solving skills and your software development skills. It is not representative of what your day-to-day work would look like at Bede and is purely for the recruitment process.

We expect that this task will take 3-4 hours to be completed, and you will be scored on your solution's **readability, testability, extensibility, and configurability**. While we would rather you didn't use AI, we will accept strategic use of LLMs where it still displays your own knowledge.

Please feel free to add any improvements you can identify along the way. However these enhancements are appreciated, they are not essential.

Please follow this spec carefully!

The Problem

You will have to build up a simplified lottery game. The solution should be a console application written in C#.

Lottery Game Mechanics

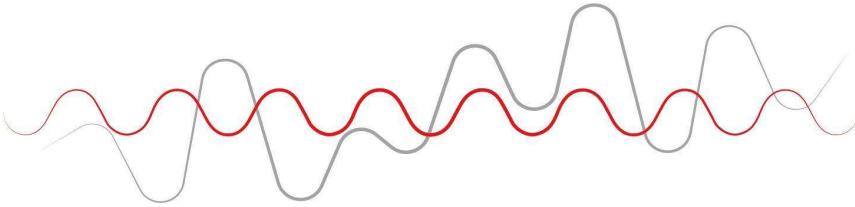
A lottery is a gambling competition in which people obtain numbered tickets, each of which has the chance of winning a prize. At a set time, the winners are randomly drawn from a pool holding all purchased tickets.

Ticket Purchase

- The user (Player 1) is prompted via the console to purchase their desired number of tickets. The remaining participants are CPU players, sequentially numbered as Player 2, Player 3, and so on. Their number of tickets is randomly picked.

Player Limits and Cost

- The total number of players in the lottery game should be between 10 and 15.
- All players (human and CPU) are limited to purchasing between 1 and 10 tickets.



- Each player begins with a starting balance of \$10.
- Tickets are priced at \$1 each.

Prize Determination

The program should determine the winners by the following rules:

- **Grand Prize:** A single ticket must be awarded a prize equivalent to 50% of the total ticket revenue.
- **Second Tier:** 10% of the tickets must share 30% of the total ticket revenue equally.
- **Third Tier:** 20% of the tickets must share 10% of the total ticket revenue equally.

Tickets selected as winners are removed from the subsequent prize drawings to prevent duplicate wins.

The remaining revenue after prize allocation is the house profit.

Note: In case the number of winners for a prize tier is not exactly divisible then the closest equal split is taken, and the remaining amount is added to the house profit.

Result Presentation

The program should output a list of the winning players, including the count of their winning tickets (in brackets next to the player identifier, e.g., {playerId} (winning tickets count)), the amount they have won, and the house profit, all printed to the console.

Sample:

```
Welcome to the Bede Lottery, Player 1!

* Your digital balance: $10.00
* Ticket Price: $1.00 each

How many tickets do you want to buy, Player 1?
5

13 other CPU players also have purchased tickets.

Ticket Draw Results:

* Grand Prize: Player 7(1) wins $50.00!
* Second Tier: Players 1(2), 3(1), 5(1), 6(1), 9(3), 11(1), 13(1) win $3.00 per winning ticket!
* Third Tier: Players 1(1), 2(1), 3(1), 4(2), 5(2), 6(1), 7(2), 8(1), 9(3), 10(1), 11(1), 12(2), 13(1), 14(1) win $0.50 per winning ticket!

Congratulations to the winners!

House Profit: $10.00
```