



# Game Theory

Presentation by Himank Gupta Khushi Tongia Shashwat Pandey





# OVERVIEW.

- **→** Introduction
- Auction and its Types
- Theory of Auctions

- Game 1
- Implementation
- Payoff

- Game 2
- Implementation
- Payoff

# Introduction

This is a research-based project centred around game theory, where we shall analyse experimental data which will be obtained using two auction games that are designed by us.





#### What is an auction?

An auction refers to the sale of goods or services by offering them up for bids

Auctions are based on the idea that competitive bidding tends to push prices higher, thus maximising profits

#### **Types of Auctions**

We shall be analysing two types of auctions:

- 1 First-price, sealed-bid auction
- 2 Second-price, sealed-bid (Vickrey) auction



# Theory of Auctions

→ First-Price, Sealed-Bid Auction

In this type of auction, all the bidders simultaneously submit sealed bids to the auctioneer.

No bidder knows the bid of any other participant.

The highest bidder wins and pays the value of their bid

#### → Second-Price, Sealed-Bid (Vickrey) Auction

In this type of auction, all the bidders simultaneously submit sealed bids to the auctioneer.

No bidder knows the bid of any other participant.

## The highest bidder wins and pays the value of the second-highest bid

These types of auctions are named after William Vickrey, an economics professor and Nobel Laureate. He was the first to use the tools of game theory to explain the dynamics of auctions.



# Game

#### BASED ON THE PRINCIPLE OF FIRST PRICE AUCTION



### IMPLEMENTATION

- The bidders can bid any amount between \$10 and \$100.
- Each player will submit 2 sealed bids per round.
- We fix the number of rounds for this game that is 10.

Winner of each round is the bidder whose average of both the submitted bids is closest to the root mean square of all the bids placed in that round.

Winner will pay an amount that is equal to the average of maximum of the 2 bids placed by the winner in all the rounds

Payoff = 
$$sum(max(bid1,bid2))/10$$

In case of a tie winner is randomly decided!

	Į.	4	В		С	
ROUND	Bid 1	Bid 2	Bid 1	Bid 2	Bid 1	Bid 2
1	99	27	60	89	38	86
2	60	61	59	59	53	56
3	32	62	64	69	63	10
4	70	88	11	63	17	90
5	45	71	24	78	80	39
6	63	98	32	32	36	50
7	78	12	96	59	84	16
8	89	85	99	72	85	35
9	27	53	79	67	42	28
10	13	39	86	28	29	54

- eg. Round 1:
   RMS of all bids = 71.75
   Closest bid = 60 (B)
   Hence, winner is B.
- Similarly we choose the winners of the other rounds.
- B wins 4 rounds,
   C wins 4 rounds,
   A wins 2 rounds.
- Thus the winner is randomly decided via coin toss between B and C to break the tie.
- Final winner: B

## PAYOFF

sum(maximum(bid1, bid2)) / 10

```
=(89+59+69+63+78+32+96+99+79+86)/10
= 75
```

Player B wins the game and will pay 75

#### Link to the spreadsheet:

https://docs.google.com/spreadsheets/d/1VBoWkL6fySd-5ZDC8T0ESz1ui8VSJfvtUwt57sWqiBk/edit?usp=sharing



# Gamez

#### BASED ON THE PRINCIPLE OF SECOND PRICE AUCTION



### IMPLEMENTATION

- The bidders can bid any amount between \$20 and \$100.
- Each player will submit 1 sealed bid per round.
- Number of rounds for this game will be log2(n), where n is the maximum bid placed in the first round.

Winner of each round is the bidder with the maximum bid

Winner will pay an amount that is 5 more than the lowest bid placed in the round number closest to [log2(n)/2].

Payoff = min(bid of round{
$$[log2(n)/2]$$
}) + 5

In case of a tie winner is randomly decided!

ROUND	A	В	С
7	60	96	29
2	28	86	26
3	95	45	25
4	93	79	88
5	87	68	56
6	92	49	64
7	99	66	95

- Highest bid in round 1 = 96
- Number of rounds = 7
   [As log2(96) = 6.58 approx]
- Winners of each round are decided by the max bid
- A wins 5 rounds,
  B wins 2 rounds,
  C wins 0 rounds.
- Final winner: A

## PAYOFF

```
min(bid of round{[log2(n)/2]}) + 5
```

- = min(bid of round 3) + 5
- = 25 + 5
- = 30

Player A wins the game and will pay 30

#### Link to the spreadsheet:

https://docs.google.com/spreadsheets/d/1ABYeUw9GXc5fOzwkF4 O6CQeXhHxtUZHhHBSNG4VY93I/edit#gid=0

# Thank You!



