UNIVERSITY CUP

ENTELECT CHALLENGE 2025

entelect
everything is possible

MegaZoo Master Planner

Introduction

Welcome to the "MegaZoo Master Planner" Hackathon! Megazoo Inc. is expanding its global footprint, and they need your brilliant mind to design the next iconic zoo. With a limited budget and a sprawling 2D grid, your task is to strategically place animal enclosures and essential visitor buildings. Your design needs to be profitable, harmonious, and optimized to maximize visitor interest in every enclosure and building.

Goal

Your goal is to develop an algorithm that strategically places a variety of **zoo resources** (animal enclosures, buildings, and pathways) within a 2D grid. We'll score your solution based on how optimally you use these resources, considering several key constraints:

- You'll need to respect the **compatibility of surrounding enclosures and buildings,** for example you can't place a predator in a certain radius of prey and certain animals can't be near a noisy restaurant.
- Your zoo should be captivating and attract visitors, which we refer to as the **“interest factor”**.
- When building your zoo, you will need to strive to keep the overall **budget as low as possible**.
- Incorporate a **diverse range of resources**, as diversity is rewarded!

When designing your zoo, you must strike a careful balance between the cost of a resource and its "interest factor". For example, a Lion enclosure will likely have a high interest factor, attracting many visitors, but it will also come with a significant cost. However, strategically placing a high-interest element like a Lion enclosure could also boost the interest factor of less captivating enclosures nearby, as the increased visitor traffic to the Lion enclosure might draw more attention to its surroundings.

The most meticulously calculated solution, balancing visitor interest with budget efficiency, will be rewarded!

Are you ready to start planning your ultimate zoo?

Zoo

The zoo is structured as a 2-dimensional grid.

For each level of the challenge, you'll receive input as a JSON representation of this 2D array in a text file. By default, every cell (or "block") in the grid is initially filled with a pathway resource. Each zoo provided for a level will also include a **list of resources** you can use to populate it.

To place a resource in the zoo, you will populate the grid cells that the resource occupies with its unique **`resource_id`**.

Below is an example of an empty zoo being populated with resources and its corresponding JSON representation that will be provided and submitted respectively:

1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1

Empty Zoo

7	7	7	1	11	11	12	12	12	12
1	7	7	1	11	11	1	1	1	12
1	7	7	11	11	11	1	1	1	12
1	7	7	1	11	1	1	1	9	9
1	1	1	1	1	1	1	9	9	9
3	3	3	1	1	1	1	1	9	9
3	3	3	1	1	1	1	1	1	1
3	3	1	1	1	1	1	1	1	7
3	3	3	1	1	1	7	7	7	7
3	3	3	1	1	1	7	7	7	7

Populated Zoo

```
{
 "level": 0,
 "zoo_size": "10x10",
 "resources": [1, 3, 7, 9, 11, 12],
 "zoo": [
  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],
  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],
  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],
  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],
  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],
  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],
  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],
  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],
  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],
  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
 ]
}
```

Empty Zoo JSON file

```
{
 "level": 0,
 "zoo_size": "10x10",
 "resources": [1, 3, 7, 9, 11, 12],
 "zoo": [
  [7, 7, 7, 1, 11, 11, 12, 12, 12, 12],
  [1, 7, 7, 1, 11, 11, 1, 1, 1, 12],
  [1, 7, 7, 11, 11, 11, 1, 1, 1, 12],
  [1, 7, 7, 1, 11, 1, 1, 1, 9, 9],
  [1, 1, 1, 1, 1, 1, 1, 9, 9, 9],
  [3, 3, 3, 1, 1, 1, 1, 1, 9, 9],
  [3, 3, 3, 1, 1, 1, 1, 1, 1, 1],
  [3, 3, 3, 1, 1, 1, 1, 1, 1, 7],
  [3, 3, 3, 1, 1, 1, 7, 7, 7, 7],
  [3, 3, 3, 1, 1, 1, 7, 7, 7, 7]
 ]
}
```

Populated Zoo JSON file

Resources

A JSON file has been provided which contains all the available resources that can be used to populate the zoo, and its structure as follows:


```
{
 "resources": [
  {
   "resource_id": 7,
   "name": "African Penguin",
   "type": "Bird Enclosure",
   "bounding_box": 4,
   "cost": 412345,
   "interest_factor": 55,
   "incompatible_with": [7, 3, 5, 27, 28, 29, 30, 31],
   "orientations": [
    {"rotation": 0, "cells": [[0, 0], [0, 1], [0, 2], [1, 1], [1, 2], [2, 1], [2, 2], [3, 1], [3, 2]]},
    {"rotation": 1, "cells": [[0, 3], [1, 0], [1, 1], [1, 2], [1, 3], [2, 0], [2, 1], [2, 2], [2, 3]]},
    {"rotation": 2, "cells": [[0, 0], [0, 1], [1, 0], [1, 1], [2, 0], [2, 1], [3, 0], [3, 1], [3, 2]]},
    {"rotation": 3, "cells": [[0, 0], [0, 1], [0, 2], [0, 3], [1, 0], [1, 1], [1, 2], [1, 3], [2, 0]]}
   ]
  }
 ]
}
```

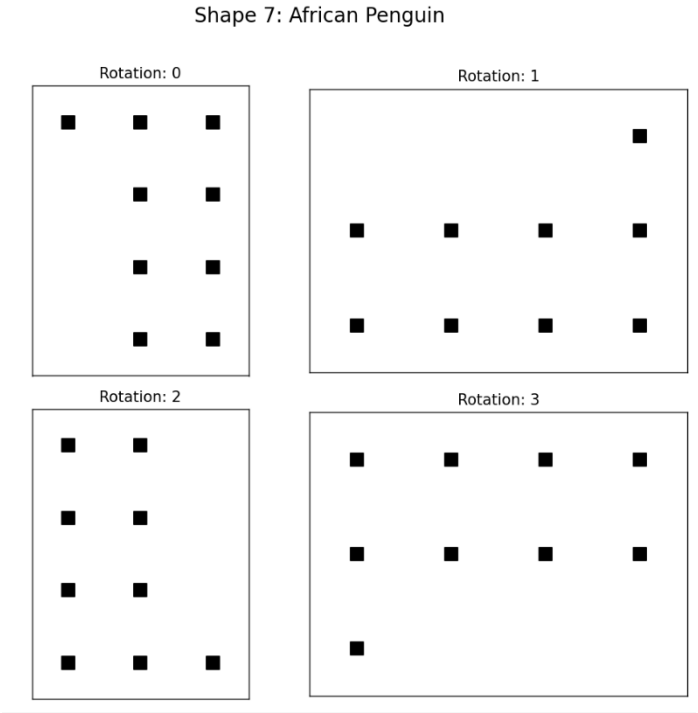

- **`resources`:** This is the top-level array that will contain a list of all distinct zoo elements available for placement. Each object within this array represents a single type of resource.
    - **`resource_id`**:
        - A unique integer identifier for each specific type of zoo element. For example, 27 identifies the "Bird enclosure".
    - **`bounding_box`**:
        - This integer indicates the size of the smallest square grid that can contain the shape of the element in any of its rotations. It helps in understanding the maximum spatial footprint of the element.
    - **`cost`**:
        - The numerical cost associated with placing this element in the zoo. This value directly impacts on the overall budget and subsequently the final score.
    - **`interest_factor`**:
        - A float value (e.g., 2) that acts as a score multiplier. Elements with higher interest factors contribute more positively to the total score, as they indicate greater visitor attraction.
    - **`name`**:
        - A descriptive string identifying the type of resource, such as "Bird enclosure", "African Elephant Enclosure", "Coffee Shop", or "Pathway".
    - **`type`**:
        - This property categorizes the element. It can be "Enclosure" (for animal enclosures), "Building" (for visitor facilities), or "Pathway".
    - **`incompatible_with`**: This is a list of `element_id` values. It specifies other zoo elements that this particular resource cannot be placed directly adjacent to (horizontally, vertically, or diagonally). For instance, a "Bird enclosure" might be incompatible with a "Lion enclosure" (if 14 were the lion enclosure ID), or even another "Bird enclosure" of the same ID.

- **`orientations`**: This is a list of objects, each describing a possible 90-degree rotation of the element's shape.
  - **`rotation`**: An integer representing the rotation in degrees (e.g., `0`, `90`, `180`, `270`).
  - **`cells`**: A list of relative **`[row, col]`** coordinates. These coordinates define the specific grid cells that the element occupies when in that particular rotation, relative to its top-left corner within its `bounding_box`.

It is important to remember to try and make use of all the different resources available to maximize your score.

Refer to **Appendix A** below, which is a visual representation of all the resources available for the zoo.

The bounding box for resource 7 will look as follows.



Constraints

1. **Resource Placement:**
    a. Zoo resources, including enclosures, buildings, and pathways, are available in various polyomino shapes and sizes. You can rotate these elements in 90-degree intervals.
    b. Importantly, resources cannot overlap one another.
2. **Compatibility Rules:**
    a. Each resource has an `incompatible_with` list in its JSON definition, specifying other resource IDs that cannot be placed within a 5-block radius (horizontally, vertically, or diagonally).
    b. By default, resources of the same ID can't be placed within a 5-block radius of each other, except for pathway segments
3. **Resource Availability:**
    a. Each level has a subset of resources that can be used for that level. Available resources can be used as many times as you want.
4. **Interest factor:**
    a. Each zoo resource has a numerical value assigned to it that represents its attractiveness to visitors. It is important because resources with a higher interest factor contribute more to your total score and can even boost the score of other resources within a 10-block radius.
5. **Budget Impact:**
    a. The total cost of your zoo design will directly affect your overall score. Higher costs will negatively impact your score, as detailed in the scoring section.

Violation Consequences:

If any of the constraints related to Resource Placement (1), Compatibility Rules (2), or Resources Availability (3) are violated for a particular resource, that resource will be treated like a pathway and not contribute to the total score of the zoo.

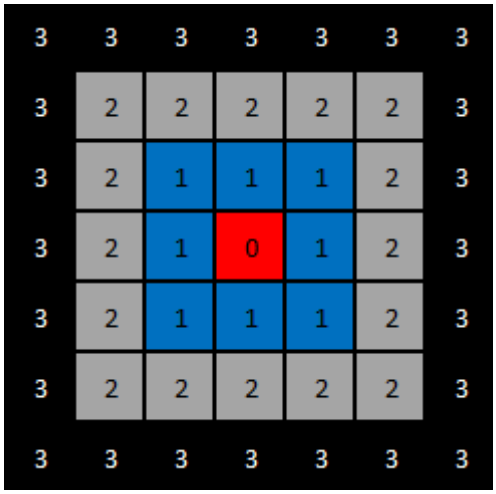
Conflict Resolution for Mutually Incompatible Resources:

To resolve placement conflicts involving bidirectional incompatibility, where two resources are mutually included in each other's **`incompatible_with`** lists, a specific validation order will be applied. If two such resources are placed within the prohibited radius of one another, the resource that is identified first during the scoring validation process will be flagged as invalid. This invalidated resource will be treated as a pathway, effectively contributing zero points to the score. The second resource will then be assessed independently and can be considered valid, provided it meets all other constraints. This "first-identified, first-invalidated" principle ensures that such conflicts are resolved predictably.

Important: The radius around a zoo resource is determined using the **Chebyshev distance**, often called the chessboard distance. So, in simpler terms the number of moves a king would take to get from one square to another on a chessboard. It's calculated as the maximum of the horizontal and vertical distances between two points.

Therefore, given two points, (x1, y1) and (x2, y2), the Chebyshev distance is calculated as follows:

max(abs(x1 - x2), abs(y1 - y2))



In this example the red square, marked **"0"**, is the starting point. The number in each surrounding square represents the minimum number of steps (or "moves") needed to reach it from that central square. This is equivalent to the number of moves a king would make on a chessboard, where it can move one square in any direction—horizontally, vertically, or diagonally.

Levels

The hackathon features four progressive levels, each building upon the complexity and constraints of the previous one to challenge your algorithm design.

- **Level 1: Basic Placement**
  - **Zoo Size:** 50 x 50 grid
  - **Constraints:** Available resources can be placed anywhere as long as they do not overlap with each other. The same resource can also **not** be placed directly next to each other (excluding pathways – ID 1); there needs to be at least a 1 block gap between the same resources if they are to be placed next to each other, otherwise the scorer will identify the resources as invalid since it can't distinguish when one resource ends and another starts if they are the same resource.
  - **Available resources:** [1, 3, 4, 6, 9, 10, 11, 14, 15, 20, 21]
  - **Objective:** Maximize the total number of resources placed in the zoo while also ensuring the diversity of resources used.
- **Level 2: Introducing Compatibility**
  - **Zoo Size:** 100 x 100 grid
  - **Constraints:**
    - All constraints from level 1 apply.
    - Resources cannot be placed within a 5-block radius of an incompatible resource.
  - **Available resources:** [1, 4, 6, 9, 10, 11, 21, 20, 2, 3, 5, 7, 8, 12, 13, 14, 15, 16, 17, 18, 19, 22]
  - **Objective:** Maximize the number of elements placed while minimizing compatibility constraint violations
- **Level 3: Introducing interest factor**
  - **Zoo Size:** 300 x 300 grid
  - **Constraints:**
    - All constraints from level 2 apply
    - Interest factors within a 10-block radius of a resource are now considered in the scoring calculation.
  - **Available resources:** All
  - **Objective:** Optimize resource placement to maximize your total score by considering interest factors.
- **Level 4: Introducing budget**
  - **Zoo Size:** 800 x 800 grid
  - **Constraints:**
    - All constraints from level 3 apply
    - Budget is now considered in the scoring calculation, therefore try keep your cost as low as possible
  - **Available resources:** All
  - **Objective:** Optimize the fine balance of cost and interest factor to maximize your score

Scoring

Each level will have its own calculation for the scoring and is described below.

Before your final score is calculated, every resource you have placed on the grid undergoes a validation process. The specific rules and checks performed depend on the level. Any resource that violates the constraints for its level is considered invalid and will contribute zero points to your score and will essentially be replaced with pathway resources which will result in it being ignored in all calculations and will effectively contribute zero points to your score. For a detailed explanation of these rules, please refer to the "Constraints" section.

Level 1& 2 calculations

Score = (**Total Valid Utilized Area**) x (**Resource Balance Multiplier**)

Total Valid Utilized Area = (Total number of cells in the grid) - (Number of cells with pathways)

Resource Balance Multiplier = (S + (1/D)) / 2

* D = Simpson's Index

* S = Unique resources count

Refer to **Appendix B** for a detailed explanation on how to calculate the balance multiplier

Level 3

Gross Score = ((Total Interest Score) x (Total Valid Utilized Area) x (Resource Balance Multiplier)) / 10000

Calculate Total Interest Score

This is the sum of the individual scores for every **valid** resource. A resource's score is its own interest factor plus a boost from nearby attractions. Pathways (`resource_id`: 1) have an interest factor of 0 and are **ignored** in this calculation.

- **Individual Score** = (`resource.interest_factor`) + (Proximity Interest Boost)
- **Proximity Interest Boost** = Σ (`neighbor_resource.interest_factor` * 0.10) for each valid neighbour within a **10-block radius**.
- **Total Interest Score** = Σ (Individual Scores of all valid resources)

The calculation of Total Valid Utilized Area & Resource Balance Multiplier is the same as in level 1 and 2.

Level 4

For the final level, the calculation takes your **Gross Score** (which is the same as the final score from Level 3) and applies a **Cost Penalty**. This means that while you still want to maximize interest, area, and diversity, you are now rewarded for doing so with a smaller budget. The lower your total cost, the smaller the penalty, and the higher your final score will be.

Level 4 Score = Gross Score / (1 + (Total Cost / 10,000,000))

The "10,000,000" is a **Budget Normalization Factor**. It's a baseline that helps turn your total cost in Rands into a fair penalty. A zoo with a total cost of R10,000,000 will have its Gross Score cut in half, while a zoo with a lower cost will be penalized less.

Given files

All required files can be found on the Entelect challenge website in the player portal under the corresponding hackathon. The following files will be provided:

One comprehensive resources JSON file that contains information for all available resources across all levels of the hackathon.

For each level, you will be provided with a text file that contains a pre-populated zoo grid for that level, available resources, level and grid size.

Submission

For each level, you must upload two items to the Entelect Challenge player portal to complete your submission:

1. **Your Populated Zoo File:** Submit a **text file** that contains the JSON representation of the zoo for that level that you have populated. Refer to the Zoo section above for an example of the format of the text file that needs to be submitted.
2. **Your Source Code:** Your complete source code must be compressed into a single .zip folder for submission.

After submitting your solution, you will receive a score along with logs that can be downloaded to better understand any errors or violations that might have occurred during the scoring of your submission.

If you receive an error, it means the JSON inside the text file you provided failed either of the following basic validations

- The JSON is malformed
- The zoo's dimensions have changed.
- Zoo object is missing

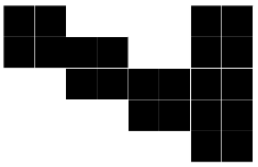
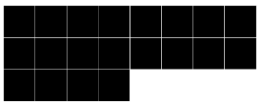
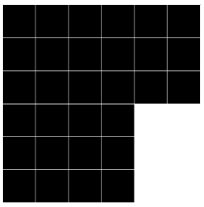
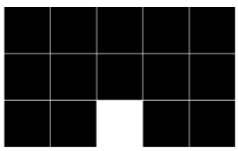
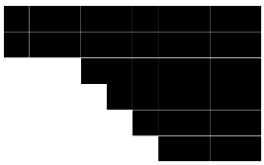
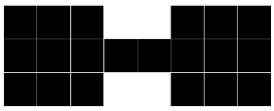
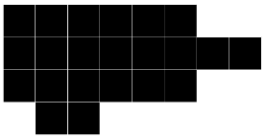
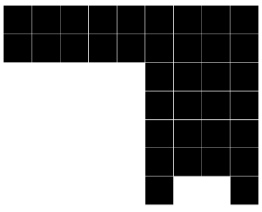
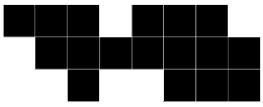
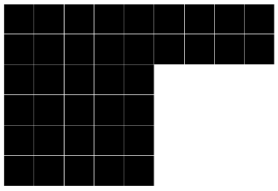
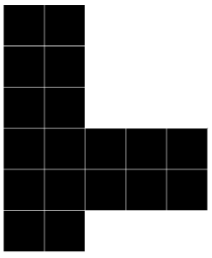
Example of level .txt file

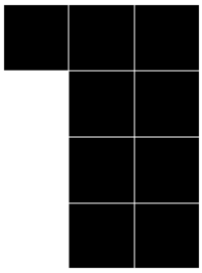
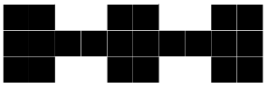
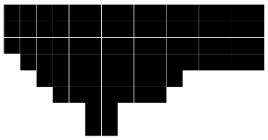
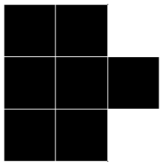
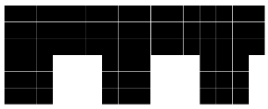
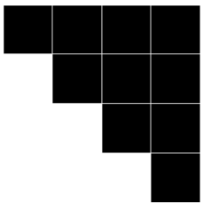
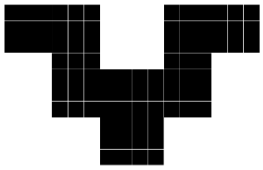
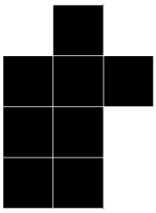
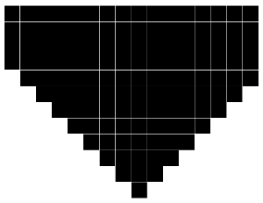
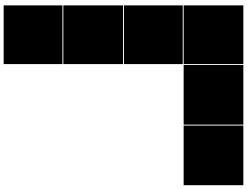
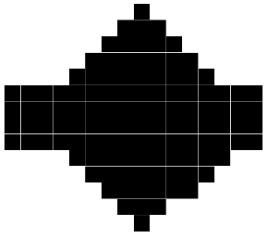
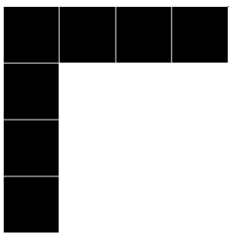
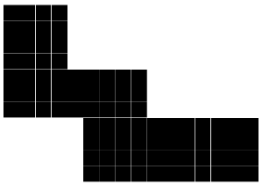
```
Level Number: 1

Zoo Size: 10x10

Available Resources: [1, 3, 7, 9, 11, 12]

Base Zoo: [

  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],

  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],

  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],

  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],

  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],

  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],

  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],

  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],

  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1],

  [1, 1, 1, 1, 1, 1, 1, 1, 1, 1]

]
```

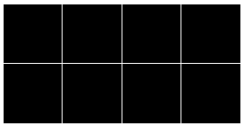
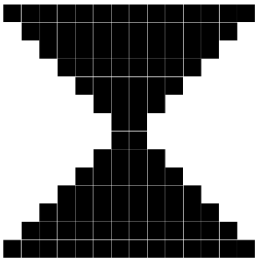
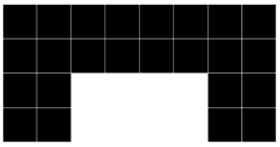
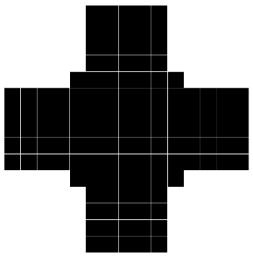
Example of submission .txt file

```
{

 "zoo": [

  [7, 7, 7, 1, 11, 11, 12, 12, 12, 12],

  [1, 7, 7, 1, 11, 11, 1, 1, 1, 12],

  [1, 7, 7, 11, 11, 11, 1, 1, 1, 12],

  [1, 7, 7, 1, 11, 1, 1, 1, 9, 9],

  [1, 1, 1, 1, 1, 1, 1, 9, 9, 9],

  [3, 3, 3, 1, 1, 1, 1, 1, 9, 9],

  [3, 3, 3, 1, 1, 1, 1, 1, 1, 1],

  [3, 3, 3, 1, 1, 1, 1, 1, 1, 7],

  [3, 3, 3, 1, 1, 1, 7, 7, 7, 7],

  [3, 3, 3, 1, 1, 1, 7, 7, 7, 7]

 ]

}
```

Appendix A

ID	Resource	Details	ID	Resource	Details
1		***name**: Pathway segment* ***type**: Building Material* ***bounding box:** 1* ***cost**: 0* ***interest**: 0* ***incompatible**: []*	16		***name**: Kudu* ***type**: Herbivore Enclosure* ***bounding box**: 8* ***cost**: R465,432* ***interest**: 55* ***incompatible**: [16, 12, 13, 14, 28, 29, 30, 31]*
2		***name**: Parking lot* ***type**: Building* ***bounding box**: 8* ***cost**: R1,523,456* ***interest**: 10* ***incompatible**: [2]*	17		***name**: Wildebeest* ***type**: Herbivore Enclosure* ***bounding box**: 6* ***cost**: R512,345* ***interest**: 60* ***incompatible**: [17, 12, 13, 14, 28, 29, 30, 31]*
3		***name**: Coffee shop* ***type**: Building* ***bounding box**: 5* ***cost**: R812,345* ***interest**: 35* ***incompatible**: [3, 9, 10, 11, 12, 13, 14, 15, 16]*	18		***name**: Zebra* ***type**: Herbivore Enclosure* ***bounding box**: 10* ***cost**: R498,765* ***interest**: 60* ***incompatible**: [18, 12, 13, 14, 28, 29, 30, 31]*
4		***name**: Bathroom* ***type**: Building* ***bounding box**: 0* ***cost**: R321,098* ***interest**: 8* ***incompatible**: [4]*	19		***name**: Baboon* ***type**: Omnivore Enclosure* ***bounding box**: 8* ***cost**: R365,432* ***interest**: 50* ***incompatible**: [19, 12, 13, 14, 28, 29, 30, 31*
5		***name**: Restaurant* ***type**: Building* ***bounding box**: 9* ***cost**: R1,276,543* ***interest**: 60* ***incompatible**: [5, 9, 10, 11, 12, 13, 14, 15, 16]*	20		***name**: Warthog* ***type**: Omnivore Enclosure* ***bounding box**: 8* ***cost**: R332,109* ***interest**: 45* ***incompatible**: [20, 12, 13, 14, 28, 29, 30, 31]*
6		***name**: Gift shop* ***type**: Building* ***bounding box**: 9* ***cost**: R734,567* ***interest**: 50* ***incompatible**: [6]*	21		***name**: Fox* ***type**: Omnivore Enclosure* ***bounding box**: 6* ***cost**: R297,654* ***interest**: 40* ***incompatible**: [21, 7, 8, 9, 10, 11, 15, 16, 17, 18, 19, 20, 22, 23, 24, 25, 26*

7		***name**: African Penguin* ***type**: Bird Enclosure* ***bounding box**: 4* ***cost**: R412,345* ***interest**: 55* ***incompatible**: [7, 3, 5, 27, 28, 29, 30, 31]*	22		***name**: Meerkat* ***type**: Omnivore Enclosure* ***bounding box**: 10* ***cost**: R215,432* ***interest**: 35* ***incompatible**: [22, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 23, 24, 25, 26, 27, 28, 29, 30, 31]*
8		***name**: Blue Crane* ***type**: Bird Enclosure* ***bounding box**: 6* ***cost**: R367,890* ***interest**: 50* ***incompatible**: [8, 3, 5, 27, 28, 29, 30, 31]*	23		***name**: Elephant* ***type**: Herbivore Enclosure* ***bounding box**: 18* ***cost**: R2,543,210* ***interest**: 95* ***incompatible**: [23, 12, 13, 14, 28, 29, 30, 31*
9		***name**: Guineafowl* ***type**: Bird Enclosure* ***bounding box**: 3* ***cost**: R21,0987* ***interest**: 30* ***incompatible**: [9, 3, 5, 27, 28, 29, 30, 31]*	24		***name**: Giraffe* ***type**: Herbivore Enclosure* ***bounding box**: 18* ***cost**: R2,087,654* ***interest**: 85* ***incompatible**: [24, 12, 13, 14, 28, 29, 30, 31]*
10		***name**: Spotted Eagle Owl* ***type**: Bird Enclosure* ***bounding box**: 4* ***cost**: R298,765* ***interest**: 45* ***incompatible**: [10, 3, 5, 7, 8, 9]*	25		***name**: Rhinoceros* ***type**: Herbivore Enclosure* ***bounding box**: 18* ***cost**: R2,234,567* ***interest**: 90* ***incompatible**: [25, 12, 13, 14, 28, 29, 30, 31]*
11		***name**: Tortoise* ***type**: Reptile Enclosure* ***bounding box**: 4* ***cost**: R234,567* ***interest**: 30* ***incompatible**: [11, 27, 28, 29, 30, 31]*	26		***name**: Hippopotamus* ***type**: Herbivore Enclosure* ***bounding box**: 18* ***cost**: R1,876,543* ***interest**: 80* ***incompatible**: [26, 12, 13, 14, 28, 29, 30, 31]*
12		***name**: Cape Cobra* ***type**: Reptile Enclosure* ***bounding box**: 4* ***cost**: R467,890* ***interest**: 65* ***incompatible**: [12, 11, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25]*	27		***name**: Lion* ***type**: Carnivore Enclosure* ***bounding box**: 20* ***cost**: R3,054,321* ***interest**: 100* ***incompatible**: [27, 7, 8, 9, 10, 11, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26]*
13		***name**: Puff Adder* ***type**: Reptile Enclosure* ***bounding box**: 4* ***cost**: R432,109* ***interest**: 60* ***incompatible**: [13, 11, 12, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25]*	28		***name**: Leopard* ***type**: Carnivore Enclosure* ***bounding box**: 20* ***cost**: R2,898,765* ***interest**: 95* ***incompatible**: [28, 7, 8, 9, 10, 11, 15, 16, 17, 18,*

					19, 20, 21, 22, 23, 24, 25, 26]
14		***name**: Boomslang* ***type**: Reptile Enclosure* ***bounding box**: 4* ***cost**: R398,765* ***interest**: 55* ***incompatible**: [14, 11, 12, 13, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25]*	29		***name**: Cheetah* ***type**: Carnivore Enclosure* ***bounding box**: 14* ***cost**: R2,787,654* ***interest**: 90* ***incompatible**: [29, 7, 8, 9, 10, 11, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26]*
15		***name**: Springbok* ***type**: Herbivore Enclosure* ***bounding box**: 8* ***cost**: R310,987* ***interest**: 40* ***incompatible**: [15, 12, 13, 14, 28, 29, 30, 31]*	30		***name**: Spotted Hyena* ***type**: Carnivore Enclosure* ***bounding box**: 15* ***cost**: R2,674,873* ***interest**: 88* ***incompatible**: [30, 7, 8, 9, 10, 11, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26]*

Appendix B

Resource Balance Multiplier

It is designed to reward you for using a wide range of resources in a balanced way, preventing the strategy of using just one of each type and then spamming a single resource. Therefore, it rewards both the variety and the evenness of your resource usage.

- **Calculation:** This multiplier is calculated using the Gini-Simpson Index. This formula calculates your multiplier by averaging your score for resource variety (S) with your score for distribution evenness (1/D), creating a balanced reward for both using many different resources and distributing them evenly.
- **Formula Steps:**
    1. **Count Unique Resource Types (S):** Count the total number of different valid resource types you have placed in the zoo.
    2. **Calculate Proportions (p_i):** For each unique resource type, count its number of instances and divide by the total number of all valid resource instances placed in the zoo.
    3. **Calculate Simpson's Index (D):** Square the proportion (p_i) for each resource type and then sum all the results. $D = \Sigma(p_i)^2$
    4. **Calculate the Multiplier:** The multiplier is calculated by averaging your score for resource variety (S) with your score for distribution evenness (1/D). Resource Balance Multiplier = $(S + (1/D)) / 2$