bmiibo.md

Bmiibo Buto Chess

A meme I took too far

Bmiibo Buto Chess is a play on words of "amiibo" and "auto chess" in a moment of genius/idiocy I was like: "I could totally make SSB amiibo fighters" but "amiibo" is already taken, so I summoned up all my intelligence, creativity, and wit to create my own new original brand.

What the fuck is a Bmiibo?

A Bmiibo is a virtual fighter in a text-based auto-battler game that I created, it uses an algorithm known as Q-learning to learn how to play the game but here's the kicker, each one is customizable. A Bmiibo's attack's and abilities can be customized using JSON or through the help of my handy bot, Judy (short for Adjudicator). This customization of fighters adds depth to the game as it's meta changes based on the people who are playing it, currently there are 9 "actionTypes" in the game (more will be added) which may not sound like a lot but the properties of these actions can be edited and multiple actions can be sequenced in different ways.

The Game

The Board

The game is played on an `NxN` board (the values of `N` are normally 4, 8, or 16) and each player's Bmiibo is placed on a random cell (the algorithm actually does its best to space them out on the board but a large number of players on a small board can result in closer starting positions).

Play

At the start of a match the order of the players is shuffled and one by one a Bmiibo takes 1 action then passes it's turn to the next. Play ends when only one Bmiibo is left.

But how is a Bmiibo removed?

That's easy, death. A Bmiibo begins each match with 100 hitpoints and when it is reduced to 0 it is considered "dead" (more on this later).

What actions can a Bmiibo take?

There are 4 actions: moving, attacking, ability, and ultimate. Moving can be done every turn but attacks, abilities, and ultimates all have cooldowns though a cooldown of 1 means it can be performed every turn. All actions that involve a cooldown are "ticked" at the start of the turn but cannot be "overticked." A Bmiibo must take an action on its turn but it is entirely possible for a Bmiibo to take an action that does not properly resolve e.g if it makes a ranged attack targeting a square that doesen't have an enemy in it the action will still be used up but will do nothing. the difference between attacks and abilities/ultimates is that attacks *must* target an enemy whereas abilities/ultimates can target any cell. The only thing that distinguishes ultimates from abilities is that the balance-formula (discussed later) allows them to do more damage/healing etc. than a regular ability.

Customizing

The files for your Bmiibos actions should be named in the format `{bmiibo name}_{attack|ultimate|ability}.json` and a Bmiibo requires one of each to work. For example here is a simple attack.

```
{
  "actionType": "melee",
  "element": "normal",
  "amount": 10,
  "cooldown": 1
}
```

The `actionType` tells us what type of action to trigger here, `cooldown` is how long we have to wait before we can use this attack again, `amount` is how much damage this melee action should do, and lastly `element` is the type of damage which potentially causes the damage dealt to be doubled if the target has a weakness to it.

An aside on elements

The elements available are "normal", "fire", "water", "earth", "air", "light", and "dark" using a particular element is often just by preference.

Customizing (cont.)

It is also possible to define an action group if you want your action to do more than one thing.

```
[{
  "actionType": "melee",
  "element": "normal",
  "amount": 10,
  "cooldown": 1
},
{
  "actionType": "heal",
  "amount": 10,
  "selfTargeting": 1
}]
```

The actions are executed in the order they appear in the `[]` and the `cooldown` of the first action is used for the entire group. Since the first action is the same as before I will only explain the second one. `actionType` tells us that this part should execute the heal action, `amount` tells us how much we should heal the target by, `selfTargeting` has a value of 1 (meaning true) and tells us that this action should target the user—this is important because this is an action for an attack so if set to 0 it would heal our enemy whereas with `selfTargeting` set to 1 we will damage our enemy and heal ourselves.

Current actionTypes

Here is a list of all the actionTypes available at this point. The basic parameters that every action must have are: `actionType` and `cooldown`

melee

The parameters for this are: `element` and `amount` . Does damage to an adjacent bmiibo equal to `amount` and of type `element` .

heal

The parameters for this are: `selfTargeting` and `amount` . Restores `amount` health up to 100 and can auto target the user if `selftargeting` is 1, if 0 targets the chosen cell.

ranged

The parameters for this are: `element` and `amount` . Does damage to a non-adjacent bmiibo equal to `amount` and of type `element` .

blockade

The parameter for this is: `blocks` . Creates an impassable (Blocked cell) adjacent to the user, up to `blocks` blocks can be present at any one time, when this amount is exceeded the block placed first is removed.

weakness

The parameters for this are: `element` , `selfTargeting` , `remove` . Adds or removes weakness to an `element` depending on whether or not `remove` is equal to 0 or 1 and targets a specific cell or the user depending on whether `selfTargeting` is 0 or 1.

explosion

The parameters for this are: `amount` , `radius` , `force` , `element` . Deals an `amount` of `element` damage to enemies in an area centered on the target cell with the specified `radius` pushing them back an amount equal to `force` from the centre. If an enemy cannot be pushed the entire distance away they do not move at all, additionally the target cell must be a distance greater than `radius` +1 away from the user.

whirl

The parameters for this are: `element` , `amount` , `force` . Melee damage of `amount` and `element` is dealt to all adjacent enemies who are each pushed away from the user by an amount `force` . Like explosion if they cannot move the full distance they are not moved at all.

piercing

The parameters for this are: `element` , `amount` . Does damage of `amount` and `element` to all enemies in the direction of target provided the target is horizontally or vertically inline with the user.

charge

The parameters for this are: `element` , `amount` , `recoil` , `distance` . Moves `distance` times towards the target and if it lands on an adjacent cell it does melee damage of `amount` and `element` but deals damage to the user of `recoil` and `element` regardless. `recoil` must be a non-zero number.

Caveats in death

Death is actually slightly more complex than I eluded to earlier, in actuality when a bmiibo drops to 0 hp it does not "die" until the start of its next turn meaning that it remains within the realm of possibility for an enemy bmiibo to to heal a dying bmiibo if it thinks this will be beneficial, additionally over damaging a bmiibo is also possible because of this though such actions are quite rare.

The Balance formula

What is the balance formula?

The balance formula is a function that checks your different actions to make sure they are "balanced" meaning they are not considered to be too powerful.

Ok but what *is* the balance formula?

So the formula works like this:

You start off with a number of points depending on the type of action, calculated as a constant multiplied by the `cooldown` for the action. The multipliers are as follows: 10 for attacks and ultimates and 5 for abilities. e.g an ultimate with a `cooldown` of 10 would start you off with 100 points.

Next points are subtracted based on the parameters of your action. The different parameters that affect the outcome are as follows:

- `amount` : the sum of all damage done by melee action components is subtracted from the total, then if any are ranged or multi-hit this is doubled. For healing this is multiplied by 1.5 and rounded to the nearest integer.
- `distance` and `force` : the value here is simply multiplied by 3
- `recoil` : points are *added* equal to this value
- `selftargeting` and `remove` (specifically for `weakness`): if `selfTargeting` and `remove` are 1 10 points are subtracted however if they're 1 and 0 10 points are *added*. On the other hand if `selfTargeting` is 0 or not present then 10 is subtracted
- `blocks` : the value here is simply multiplied by 10

Some additional restrictions exist such as the maximum amount of damage an attack, ability and ultimate can do (20, 40, 70 respectively) and the minimum cooldown for ultimates and abilities (10 and 3 respectively). For the `blockade` action there is a maximum value of 2 for the `blocks` parameter and 1 occurrence of the action component per action (this is to prevent a situation where a Bmiibo could "turtle up" and use ranged attacks to guarantee a win). Lastly `recoil` must at least have a value of 5.