

```

1  #pragma once
2  #include "GameInstance.h"
3  #include "PlayerConsole.h"
4  #include "PlayerRandom.h"
5  #include "PlayerCBR.h"
6  #include "Player.h"
7  #include "PlayerDeterministic.h"
8  #include "Pokemon Final\CBRCasBaseLinear.h"
9  #include "Pokemon Final\CBRCas.h"
10 #include "Pokemon Final\CBREnviroment.h"
11 #include "PokemonMareep.h"
12 #include "PokemonMagikarp.h"
13 #include <iostream>
14 #include <chrono>
15 //
16 void TestAIInteraction();
17 //This test checks whether the save load mechanics work
18 void TestCaseSaveLoad();
19 //This plays a single AI against a random oponent 2000 times, loading from preset
  weights
20 void TestPlayOverTime();
21 //This plays a single AI against a random oponent 2000 times, loading from preset
  weights
22 void TestPlayDetermanisim();
23 //
24 void TestMergeSort();
25 //
26 void TestKNN();
27 //
28 void TestCaseAdaption();
29 //Times AI turns
30 void TestTurnTime();
31
32 static void TDisplayConsole(GameInstance * gi, std::unique_ptr<Player> * Players,
  GameInstance::MovePairs moves);
33 float TPlayCBRvsRandomInstance(std::unique_ptr<CBRInstance> * AI, int gamemax, bool
  DisplayRoundP = false);
34 template<class PlayerClass> float TPlayCBRvsPlayer(std::unique_ptr<CBRInstance> *
  AI, PlayerClass * player, int gamemax, bool DisplayRoundP = false);
35 float TPlayCBRvsDeterministicInstance(std::unique_ptr<CBRInstance> * AI, int
  gamemax, bool DisplayRoundP = false);
36 float TPlayCBRvsConsoleInstance(std::unique_ptr<CBRInstance> * AI, int gamemax);

```