<u>Pokemoz:</u>

<u>Context:</u>

The game *Pokemoz* was designed for the class LINGI1131*.* We have followed the project instructions but also added a personal touch.
In order to stick to reality a little more, the wild Pokemoz's evolve through the game and can acquire experience during combat.
After combat, if his Pokemoz is wounded or dead, the head trainer has to go back to his starting point, his house, to cure it or bring it back to life.

The artificial intelligence provided to our head trainer is *naive*: it simply consists of making his Pokemoz evolve to the tenth level through various fights, while curing it when it's been hurt or lost a fight.
Once the tenth level completed, the trainer takes the shortest way to reach the exit.

<u>Structure of the code and implementation choices:</u>

First, the structure of the three main elements will be presented: the Pokemozs, the trainers and the maps.
We have represented the Pokemozs with a record that has the following structure
$$\textbf{Pokemoz = p(type:\_ name:\_ hp:\_ lx:\_ xp:\_)}$$
Where
   type: atom that can take 3 different values : grass/fire/water
   name: string representative of the name of the pokemoz
   hp: int representative the health status of the pokemoz
   lx: int representative the level of the pokemoz
   xp: int representative the experience of the pokemoz

We have represented the trainers with a record that has the following structure
$$\textbf{Trainer = t(p:\_ x:X y:Y handle:\_ type:wild/persoPrincipal name:Name)}$$

Where
      p: PortObject of the trainer's pokemoz
      x: int representative of the abscisse of the position of the trainer on the map
      y: int representative of the ordinate of the position of the trainer on the map
      handle: graphics components of the trainer
      type: atom that can take 2 different values : wild or persoPrincipal
      name: String representative of the name of the trainer

Maps are represented as matrices built with a tuple of tuples.
The road map's matrices are filled with zeroes or ones. A value of 1 represents the fact that there is tall grass at that position while a value of 0 means that a road passes through that position.

The trainers map contains values of either 0, 1 or 1000. A value of 1 represents that a wild trainer stands on the position, a value of 1000 means that the head trainer is there and a value of 0 represents the fact that there is no trainer on the position.

In order to design the game, 4 types of PortObject were used. We have included the Pokemozs and the trainers in PortObjects to be able to update them and accede to their information during the events (fights, displacements, treatments …) that they live throughout the game.

We have decided to use three type of maps. The first one is the visual representation of the trainers and their situation: it's with this map that the player interacts.
The second one is used to represent the grass areas on the roads. The third and last one virtually represents the position and displacements of the trainers and is included in a PortObject because it is frequently updated and questioned about its state.

Finally, we used a fourth PortObject enabling us to stop the displacements of the trainers while a combat is running and to start one combat at a time.

We invite you to take a look at the component and state diagrams to understand the interactions between the PortObjects.