

MSc OO Design and Programming

1st Tutorial Adventure Game in UML

Consider a simplified adventure game:

The game takes place in a collection of rooms, which may be connected with each other through doors. The participants of the game are plants and heroes. The participants are in certain rooms. The game is played for a number of rounds. At each round each participant is given his turn. When a plant is given its turn, it increases its nutritional value. When a hero is given his turn, his strength decreases; he may also eat a plant, or fight with another hero and/or go to another room.

In more detail:

Each room has four sides (S, E, N, W). Each side is either a wall, or a door. Each room may contain several participants or none at all.

A door connects two rooms with each other.

Plants have a name, and have a nutritional value, which is initially 8.60. When a plant is given its turn, it increases its nutritional value by 0.4. When a plant is eaten, its nutritional value decreases by $\frac{1}{3}$.

Heroes have a name, and they are born with an initial strength of 12.50. When a hero is given his turn, his strength decreases by 0.4. Then he “looks around” for another hero or a plant in the room he is in. If it is a plant, then he eats it and his own strength increases by 0.8. If it is another hero, then they fight, and both their strengths decrease by 1.3. (If there is more than one participant in the room, the hero may pick any of them – i.e. in your program you may choose any strategy for that). Finally, the hero moves into a direction. If by moving into that direction he encounters a wall (i.e. he moved to the south and the southern part of his room was a wall), then his strength decreases by 0.2 (because he banged his head) and he stays in the same room; if he encounters a door, he moves into the room connected through that door. Because he cannot “see” the walls and the doors, he moves into the four directions in turn.

The first time he is given a turn he tries to go east, next time he is given a turn he tries to go south, next time he tries to go west, next time north and so on. The order he tries the directions is not affected by whether he succeeded in moving into a new room or not.

Develop an UML class diagram to describe the above. Remember that a UML diagram can usually *not* describe *all* aspects of a problem.