Hero of the Core prototype

You work in the upcoming amazing game called **Hero of the Core**, which intends to reinvent the Multiplayer online battle arena (MOBA) genre. The game will put you in a 2D arena map full of towers and enemy heroes. As one of the last members of your clan you should defend your core at all cost besides your comrades, destroying the enemy heroes and their towers.

The main designer has requested to develop a small prototype to show the base game mechanics to some investors.

In this early prototype the objective will be to test the movement and destroy mechanics against the towers and the enemy heroes, but there will be only one player hero and your core it will be safe as the enemy heroes won’t move.

Each time you destroy a tower you will win a weapon to destroy an enemy hero, if you try to defeat an enemy hero without a weapon you will be defeated! So the number of enemy heroes and towers need to be the same.

The game will end with victory when all the towers and enemies are defeated, but it will end with a defeat if your hero perishes against an enemy hero.

**Input**

The input to your game will be:

* The size of the arena given as maximum rows and columns.
* Several positions in the map, each of them with only one of the following elements:
  + Empty - represented as '\_', by default all arena is empty.
  + Hero player - represented as 'h', in this prototype it has to be only one.
  + Core - represented as 'c', it has to be only one!
  + Enemy Tower - represented as 't'
  + Enemy hero – represented as 'e'
* A dot, this will help to delimitate arena setup from movements.
* A sequence of movements given as a pair (±1, ±1), indicating the rows and columns to advance each round.
  + Your hero can only walk one position at a time in any direction.
  + If your hero walks into an enemy tower position, the tower will be destroyed.
  + If your hero walks into an enemy position, it will beat him only if it has a weapon!

3 3 🡨 this is a 3x3 arena

0 1 h 🡨 row 0 and column 1 is the hero position

0 0 c 🡨 row 0 and column 0 is the core position

1 1 e 🡨 at row 1 and column 1 there is an enemy hero

1 2 t 🡨 at row 1 and column 2 there is a tower

. 🡨 From now on all inputs are movements!

1 1 🡨 Go down one row and right one column

0 -1 🡨 Go left one column

**Output**

The output will be a matrix, showing each element and the evolution of the movements.

Game Status:

|  |  |  |
| --- | --- | --- |
| \_ | h | \_ |
| \_ | e | t |
| \_ | \_ | \_ |

[Enemies left=1][Current Weapons=0]

Game Status:

|  |  |  |
| --- | --- | --- |
| \_ | \_ | \_ |
| \_ | e | h |
| \_ | \_ | \_ |

[Enemies left=1][Current Weapons=1]

Game Status:

|  |  |  |
| --- | --- | --- |
| \_ | \_ | \_ |
| \_ | h | \_ |
| \_ | \_ | \_ |

[Enemies left=0][Current Weapons=0]

Game Status:

You have cleaned the battle arena! Congratulations!

## Example2 (death):

**Input**

3 3 🡨 this is a 3x3 arena

0 1 h 🡨 row 0 and column 1 is the hero position

0 0 c 🡨 row 0 and column 0 is the core position

1 1 e 🡨 at row 1 and column 1 there is an enemy hero

1 2 t 🡨 at row 1 and column 2 there is a tower

. 🡨 From now on all inputs are movements!

1 0 🡨 Go down one row

**Output**

Game Status:

|  |  |  |
| --- | --- | --- |
| \_ | h | \_ |
| \_ | e | t |
| \_ | \_ | \_ |

[Enemy heroes left=1] [Current weapons=0]

Game Status:

|  |  |  |
| --- | --- | --- |
| \_ | \_ | \_ |
| \_ | h | t |
| \_ | \_ | \_ |

[Enemy heroes left=1] [Current weapons=0]

You died!

## Example3 (no hero):

**Input**

3 3 🡨 this is a 3x3 arena

1 1 e 🡨 at row 1 and column 1 there is an enemy hero

1 2 t 🡨 at row 1 and column 2 there is a tower

. 🡨 From now on all inputs are movements!

1 1 🡨 Go down one row and right one column

0 -1 🡨 Go left one column

**Output**

Error: It has to be a hero and a core!

## Example4 (bounded move):

**Input**

3 3 🡨 this is a 3x3 arena

0 1 h 🡨 row 0 and column 1 is the hero position

0 0 c 🡨 row 0 and column 0 is the core position

1 1 e 🡨 at row 1 and column 1 there is an enemy hero

1 2 t 🡨 at row 1 and column 2 there is a tower

. 🡨 From now on all inputs are movements!

2 2 🡨 Go down 2 rows and right 2 columns, this is a non-allowed movement!

1 1 🡨 Go down one row and right one column

0 -1 🡨 Go left one column

**Output**

Game Status:

|  |  |  |
| --- | --- | --- |
| \_ | h | \_ |
| \_ | e | t |
| \_ | \_ | \_ |

[Enemies left=1][Current Weapons=0]

Can't move that distance! (2,2)

Game Status:

|  |  |  |
| --- | --- | --- |
| \_ | h | \_ |
| \_ | e | t |
| \_ | \_ | \_ |

[Enemies left=1][Current Weapons=0]

Game Status:

|  |  |  |
| --- | --- | --- |
| \_ | \_ | \_ |
| \_ | e | h |
| \_ | \_ | \_ |

[Enemies left=1][Current Weapons=1]

Game Status:

|  |  |  |
| --- | --- | --- |
| \_ | \_ | \_ |
| \_ | h | \_ |
| \_ | \_ | \_ |

[Enemies left=0][Current Weapons=0]

Game Status:

You have cleaned the battle arena! Congratulations!

*Example5 (several players):*

**Input**

3 3 🡨 this is a 3x3 arena

0 1 h 🡨 row 0 and column 1 is the hero position

0 0 h 🡨 row 0 and column 0 is the hero position, it should report an ERROR!

1 1 e 🡨 at row 1 and column 1 there is an enemy hero

1 2 t 🡨 at row 1 and column 2 there is a tower

. 🡨 From now on all inputs are movements!

1 1 🡨 Go down one row and right one column

0 -1 🡨 Go left one column

**Output**

Error: It can't be several main heroes!

Simplified version (without movements)

Earthcraft prototype

You work in the upcoming amazing sandbox game called **Earthcraft**. The game will be played from an overhead perspective in a 2D map and it will consist on the recollection of materials to build all kind of things while you avoid enemies that will attack you.

The main designer has requested to develop a small prototype to show the base game mechanics to some investors.

At this early stage of the prototype your task is just to create the map, setting the player, materials and enemies on the right location, taking note of the total of materials in the map.

**Input**

The input to your game will be:

* The size of the map given as maximum rows and columns.
* Several positions in the map, each of them with only one of the following elements:
  + Empty - represented as '\_', by default all map is empty.
  + Player - represented as 'p'.
  + Material - represented as 'm'
  + Enemy - represented as 'e'
* A dot, this will help to delimitate when all positions are set.

3 3 🡨 this is a 3x3 map

0 1 p 🡨 row 0 and column 1 is the player position

2 0 e 🡨 at row 2 and column 0 there is an enemy

1 2 m 🡨 at row 1 and column 2 there is a material

1 1 e 🡨 at row 1 and column 1 there is an enemy

. 🡨 Map is finished!

**Output**

The output will be a matrix, showing each element on the correct position and the number of materials.

|  |  |  |
| --- | --- | --- |
| \_ | p | \_ |
| \_ | e | m |
| e | \_ | \_ |

Materials to collect: 1

*Example2*

**Input**

2 2 🡨 this is a 2x2 map

0 0 p 🡨 row 0 and column 0 is the player position

0 1 e 🡨 at row 0 and column 1 there is an enemy

1 1 m 🡨 at row 1 and column 1 there is a material

. 🡨 Map is finished!

**Output**

|  |  |
| --- | --- |
| p | e |
| \_ | m |

Materials to collect: 1