Detecting Helper Functions

Below is a function used in a sample videogame, which contains a lot of repeated code:

```plaintext
# next-state-tick : GameState -> GameState
fun next-state-tick(g):
    if g.playerx < -50: game(700,
        g.playery,
        g.dangerx,
        g.targetx,
        g.score)
    else if g.dangerx < -50: game(g.playerx,
        g.playery,
        700,
        g.targetx,
        g.score)
    else if g.targetx < -50: game(g.playerx,
        g.playery,
        g.dangerx,
        700,
        g.score)
    else: g
end
end
```

Write the contract and purpose statement for a helper function you could write in order to make the code shorter and easier to read.

```
# ____________ : ____________________ -> __________
```

```
____________________________________________
____________________________________________
```

```
____________________________________________
____________________________________________
```
Below is a function used in a sample videogame, which contains a lot of repeated code:

```csharp
# draw-state : GameState -> Image

fun draw-state(g):
    if g.level == 1: put-image(PLAYER-IMG1,
                               g.playerx, g.playery,
                               BACKGROUND)
    else if g.level == 2: put-image(PLAYER-IMG2,
                                    g.playerx, g.playery,
                                    BACKGROUND)
    else if g.level == 3: put-image(PLAYER-IMG3,
                                    g.playerx, g.playery,
                                    BACKGROUND)
    else: g
end
end
```

Write the contract and purpose statement for a helper function you could write in order to make the code shorter and easier to read.

```
# ________________ : ___________________________ -> __________
```

```
____________________________________________
____________________________________________
```

```csharp
```