

Project: Dorc Slayer

by Christine Laurendeau



This is a *practice* project that we will be working on together in class. It will NOT be graded.

Problem Statement

King Aesop's realm has been ravaged by hordes of rabid dwarf orcs (dorcs) that have been terrorizing the populace. The King's only hope is to obtain a rare baby Dragon and train it to defend the realm. He assigns this task to Timmy Tortoise, adventurer extraordinaire, and his unshakable sidekick, handsome Prince Harold the Hare. Their goal is to climb to the top of the Mount of Doom, kidnap the baby Dragon that lives there, and make it down the mountain alive with the Dragon.

The *Dorc Slayer* game models Timmy and Harold's death-defying mission. The player directs Timmy's movements, and Harold is shown following along within a few steps of Timmy. The game generates various caves and paths for our heroes to explore and treasure that they can collect, including jewels, weapons and food. Timmy and Harold must also fight off the dorcs that are making their way down the mountain to attack our heroes. The dorcs appear randomly at the top of the mountain and can only move downward and sideways, but not up. Our heroes maintain a health indicator, which decreases when they are hit by a dorc and increases when they eat food. They begin the game with little food, and must collect some in order to stay alive. Timmy and Harold can use weapons to fight off and kill dorcs, and the effect of a weapon on a dorc is proportional to its strength. If Timmy and Harold reach the top alive, the baby Dragon might attack them, since it possesses some dorc characteristics. However, once kidnapped, the Dragon starts exhibiting hero traits instead, and it can collect treasure and attack dorcs. The game ends when either Timmy Tortoise is killed, or when Timmy reaches the bottom of the Mount of Doom with the baby Dragon.

Other features available to the player include the ability to save a game and to resume a saved game.

A system administrator menu allows a game administrator to set the frequency of dorc generation, and the speed of dorc movements, with respect to the heroes' movements.

Technical Specifications

Platform

The SCS Linux network will be used as the test bed for the *Dorc Slayer* system, so the game **must** work on this platform. All source code must be written in C++.

User Interface

Paths and the entrance of caves will be visible at all times, except when our heroes are inside a cave. The layout of each cave will only become visible when the heroes enter the cave.

The first incarnation of Dorc Slayer will be console-based, however it should be extensible to any UI platform, including graphical, with little modification.