Jumping Koala:

Makefile Madness 2

(for certain values of 2)

Prancing Koala Developers:

- ➢ Bryan Ames
- ➢ Soren Bjornstad
- ➢ Kalpit Modi

Contents

1. Introduction

Jumping Koala: Makefile Madness 2 is a physics-based platformer game in which you, playing one of several possible koala characters, attempt to jump up a series of Makefile dependencies and eventually defeat the evil g++ sorcerer at the top.

Levels are generated based on real Makefiles, with listed dependencies becoming blocks. The game comes with a number of Makefiles for you to play with, but you can use any valid Makefile to generate a level within the game. (Invalid Makefiles will not work any better for generating levels than they do for compiling software. However, they might still be useful for impressing non-technical users.) Of course, levels not explicitly approved by Prancing Koala are not guaranteed to be fun, but they should be playable at a minimum.

Rather than running and jumping as in some similar games, you move by slingshotting yourself between platforms. When you hit any side, top, or bottom of a file (block), you will stick to it, and you can then continue from there. If you continue successfully, you will eventually reach the top of the dependency tree, where the blocks will end and the evil g++ sorcerer, whom you must hit to win the game, will be visible. Just be careful not to fall or slingshot yourself too hard—even koalas can get hurt!

2. Backstory

In this game, you play as a koala of the jumping description. Your pretense for flinging yourself up the enormous tower of makefile targets is related to your diet. Koalas are notorious for eating eucalyptus leaves to the point of exclusivity. It is a lesser known fact that this food source has a mind altering effect on koalas. In this case, the protagonist has consumed too much eucalyptus, and his (or her) vision is clouded by it. What you see on screen is a metaphor for the daily struggle of the koala: namely, climbing a tree to find the freshest (and most potent) eucalyptus.

As you climb this tree of dependencies, you will face many dangers, and learn about the evil conspiracy against you and your desire to be a peaceful

eucalyptus-eater. You will become a skillful koala tosser, and when you reach the top, you will be ready to face the final challenge. You will discover the root of the problem at the canopy of the makefile diagram and defeat the evil g++ sorcerer.

This sorcerer is notorious throughout the forest for preventing koalas everywhere from eating their favorite food. His evil powers stem from his ability to harness the mistakes of his opponents. He identifies your errors and missteps, turns them into several megabytes worth of condescending error messages, then psychically transmits them directly into your brain. These serve to confuse you (or any koala he finds it necessary to attack) and subsequently make you fall from your perch all the way to the forest floor.

If you are able to complete your journey and claim your rightful position on top of the diagram, you may even learn more secrets about the identity of the g++ sorcerer
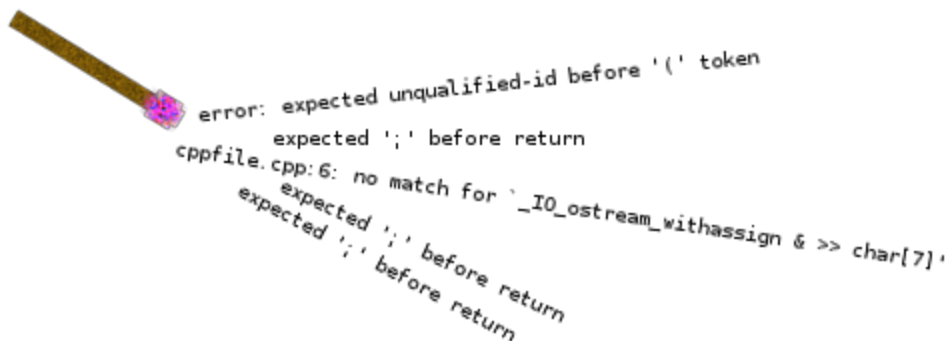
and the details of his evil plot...and of course, gorge yourself on those precious eucalyptus leaves at the top!

3. Starting and Choosing a Level

It is possible to start the game via a terminal command or by simply double-clicking the *jumpingkoala* executable file. Doing so will open a start screen that allows you to choose various options to customize your gameplay. The first button will include a button labeled "Load File" that allows you to load any makefile that you

already possess and to use that makefile to generate blocks based on the listed dependencies. We at Prancing Koalas believe that beauty is subjective for an
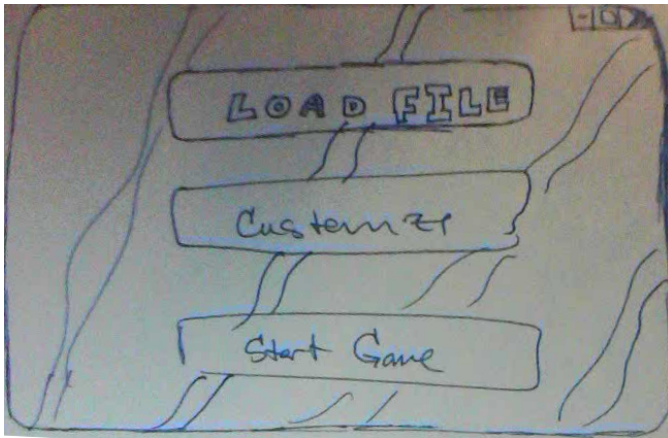
individual. As a result, we have included a second button on the start screen labeled "Customize" that opens a window, which allows you to customize your avatar to make it more visually appealing. We have added a range of options, including changing your gender, race, class, and the color of your avatar. We have also included a check-box to set the mode of the sling-shot for the duration of the game. This check box allows you to choose whether your koala avatar sticks to the platform sides or bounces off the sides when launched with the slingshot. The third and final button on the start screen is labeled "Start Game" and is intended to allow you to commence playing the level that you have selected.

4. Gameplay

Starting the game

When the makefiles are loaded and the avatar and mode customized, you are ready to click on the "Start Game" button. The screen after printing the story of your avatar will open up to the actual game screen. Your avatar will be found at the bottom rung of the makefile diagram blocks. The goal of the game is to guide your avatar up through the maze of blocks by slingshotting it along and eventually hit the g++ sorcerer on the top to win the game.

Controls

To slingshot your avatar, you can use the functions of both the mouse and the keyboard. The mouse is used to control the trajectory of your avatar.
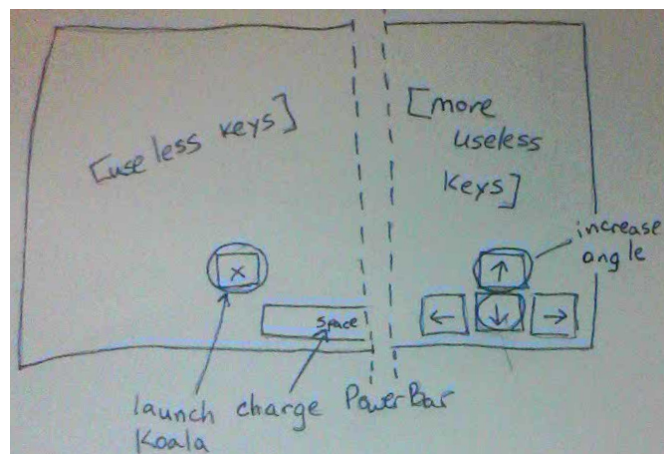
To set the trajectory of your avatar, move the mouse and you will see a faint line that describes your avatar's predicted trajectory. Alternatively, use the up and down arrow keys to move the line. Once you have found an appropriate trajectory to land on the first block, press the spacebar. The longer you hold the spacebar the harder your avatar flies. Once you have selected the speed of your avatar, you can launch it by pressing the "X" key on your keyboard. If you're playing the game in sticky mode, the avatar will cling onto all four sides of a block if it comes in contact with the block. In bouncy mode, the avatar will bounce off the sides of the block if it comes in contact with the block. You can only stop on the top of a block in bouncy mode. In all modes, as soon as you make your initial jump, the floor will fill up with water and you will no longer be able to safely land there.
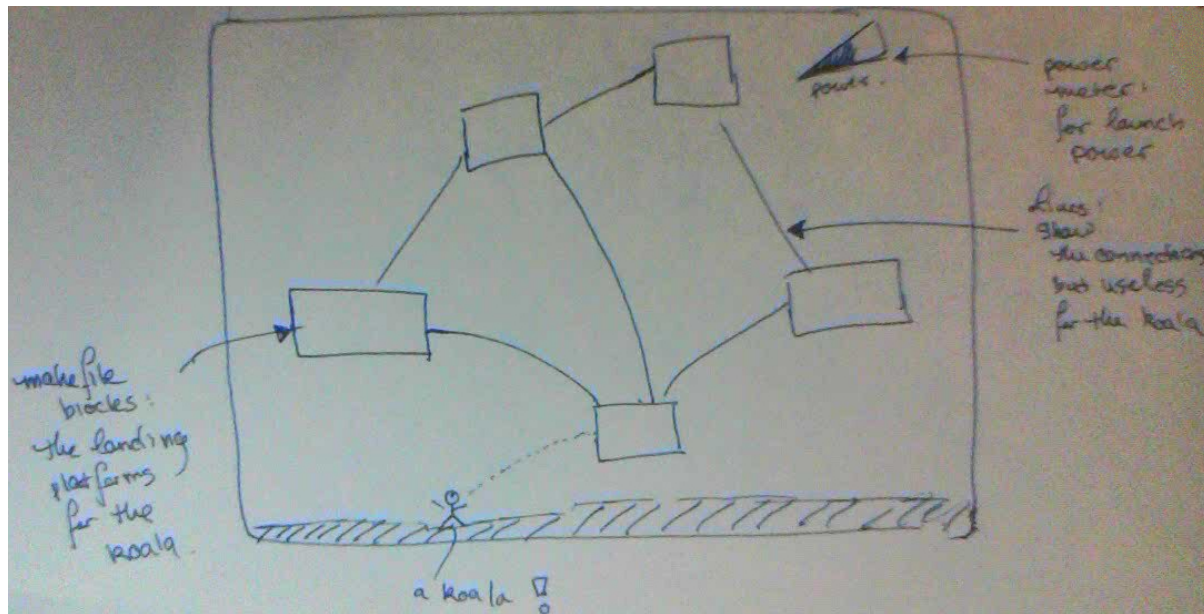
Winning

To win a level of jumping koala, you must defeat the g++ sorcerer on the topmost rung. The g++ sorcerer is in constant horizontal motion on his platform; therefore, you must perfect the trajectory, the power and the launch moment so that your avatar collides with the g++ sorcerer. If you fail to collide with the character and either overshoot or undershoot the sorcerer, your avatar will start to fall. If it hits a block on the way down, it will cling to that block. However, if it falls all the way and hits the water at the bottom, you lose the game. Hitting the

sorcerer marks a win and you are returned back to the start screen to load a different makefile, customize gameplay options, or quit.

5. Quitting the Program

Since *Jumping Koala: Makefile Madness 2* is such an awesome game, it is very difficult to quit playing it. We firmly believe in designing our games to parallel human thought, so even after you have decided to quit playing the game, we make it difficult to actually terminate the program. A simple *quit* button is so 2013! In order to close *Jumping Koala: Makefile Madness 2*, you must perform the following three steps:

1. Select the red "X" (the "quit" option) from the main menu.
2. Click on the *moving* "quit" button on the ensuing quit screen.
3. Correctly answer the mathematical question provided, which will ask you to determine the day on which a random date in the past fell. If you don't know, you'll have to figure it out first!

These steps additionally ensure that you are fully qualified to stop playing and return to daily living by checking that you have a full complement of logical, motor, and mathematical skills. Leaving the game without taking time to acclimate oneself to the real world can be dangerous! Know your limits: if you find you are not ready to leave the game, do not attempt to circumvent the quit restrictions.

The developers at Prancing Koala wish to make it clear that we consider it dangerous and immoral to cheat in the process of quitting the program, by means including but not limited to:

➢ Asking other humans and/or koalas to perform one or more of the required tasks for you.
➢ Writing or downloading software to automatically perform one or more of the tasks.

➢ Using calculators, webapps, or other automated means to perform part or all of the mathematical task.
➢ Leaving the program running while attending to other tasks in order not to have to run the quit sequence, despite not intending to play the game further in the near future.
➢ Killing the *Jumping Koala: Makefile Madness 2* process by external means without performing all of the tasks.
➢ Power cycling your computer.

Anyone who is found to have cheated in the process of quitting the program may be subject to civil and/or criminal penalties as outlined in the LICENSE file.

Naturally, this means that if you don't know how to perform the mathematical task *now*, you would be wise to use Google to look up information on how to do it before you are tired from playing the game and find yourself unable to quit the game. Similarly, if you are uncertain of your ability to click a moving quit button, you should consult your local motor skills health professional before playing *Jumping Koala: Makefile Madness 2.* If you're uncertain of your ability to locate and click a *static* quit button, this game is not appropriate for you.

6. Troubleshooting
If the game doesn't work, finish writing it first.

7. Credits and License
Jumping Koala: Makefile Madness 2 (for certain values of 2) was developed by the Prancing Koala team for the CS 251 final project at St. Olaf College.

Programming:
➢ Bryan Ames
➢ Kalpit Modi
➢ Soren Bjornstad

Sound And Music:

    TBA

Art and Other Graphical Assets (so far…):

- ➢ Bryan Ames
- ➢ Kalpit Modi
- ➢ The Internet

Special Thanks:

- ➢ Colin Scheibner, for letting us steal his idea, and then being okay with us letting it die as soon as we thought of something better.
- ➢ *Nintendo Corporation of America Inc.* for allowing us to use their [user manual](http://goo.gl/wq5Ehr) (http://goo.gl/wq5Ehr) as a model.

For details about the licensing of the game, please refer to the LICENSE file included in the root directory of the game's files.