

# robotfindskitten v1600003-201b

---

Leonard: In Bakersfield they have an art gallery in the mall.  
Al: That's where they belong.

by Leonard Richardson

leonardr@segfault.org My site: <http://www.crummy.com/> My software: <http://www.crummy.com/devel>

=====

---

Copyright © 1997, 1999, 2000 Leonard Richardson.

Permission is granted to make and distribute verbatim copies of this manual provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this manual under the conditions for verbatim copying, provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this manual into another language, under the above conditions for modified versions, except that this permission notice may be stated in a translation approved by the Foundation.

# 1 The Niceties

Hi there. This is the documentation file for robotfindskitten, winner (by default) of the first Nerth Pork webzine robotfindskitten contest. The object of the contest was to create a work of art whose title and/or theme was "robotfindskitten". I chose to write this program, which is a Zen simulation. That's what Nerth Pork editor Pete Peterson calls it, and that's really the best way to describe it, even though "Zen simulation" sounds kinda pretentious. Anyway, let's dispense with the niceties and get on with the documentation.



## 2 The Documentation

In this simulation, you play the part of robot. Your task is to complete the simulation by finding kitten, as is your destiny, and indeed your wont. You (robot) are represented by the # character, and you move around with the arrow keys touching things. If the thing you touch is kitten, you get a cute little animation (which was cuter in the DOS version) and the simulation ends. Otherwise, you get a brief description of what it is you touched. So far, there are 201 non-kitten objects in the simulation, thus the subversion number 201. Future versions will have more objects and thus higher subversion numbers.

### 2.1 The Parts of the Simulation

The first part of the simulation is the command line used to invoke it. It is here where you can control the number of non-kitten objects on the screen in any one instance of the simulation. To exercise this control, simply enter a number between 0 and 201 inclusive on the command line. For instance, to play with 32 objects on the screen, you would invoke the program with the command `robotfindskitten 32` (or, more likely, `./robotfindskitten 32`). The default is twenty non-kitten objects.

The second part of the simulation is the title screen. It should be fairly self-explanatory.

The third part of the simulation is the actual simulation itself. The long line near the top of the screen separates the screen into the field of play and the status bar. The status bar displays messages and item descriptions. Just so you don't wonder what that line is.

The fourth part of the simulation... there isn't one. It's not a very complicated simulation.

### 2.2 Controls

1. To move robot, use the arrow keys. Diagonals now work! Now that's innovation!
2. To touch an object, make a move as if you were trying to occupy the square currently occupied by the object you want to touch. Sorry if that seemed obvious, but you never know. I don't want to be accused of writing incomplete documentation.
3. If you want to end the program for whatever reason, just hit Escape, or terminate the program by hitting control-C.



## **3 A Word of Warning**

Danger!



## 4 Another Word of Warning

The characters you see on the screen are not guaranteed to correspond in shape or color to what the descriptions say they are. In fact, they're pretty much guaranteed not to. If there was any consistency, it'd be pretty easy to spot kitten every time, now wouldn't it?



## 5 Yet Another Word of Warning

This is a BETA release of robotfindskitten. I've never had it crash, but it might do weird things. Let me know.



## 6 Technical Information for Hard-Core Geeks

The original robotfindskitten (the one that won the contest) was written using Borland Turbo C++ 3.0 for DOS in 1997. I ported it to (rewrote it for, more like) Linux over Independance Day, 1999. It now uses the ncurses library. Version 1600003 is written for ncurses 5.0, which is what I have on my Mandrake system. It's good that you're a hard-core geek like me. I can speak frankly to you, knowing that non-coders won't read this section, or will be scared off by all that jargon in the first few sentences. Our plan is nearly complete, and we are poised to take our rightful places as rulers of this planet. Already we control the technology of the sheeplike masses. We merely await the command from Geek Central, and Earth will be ours! AH HA HA HA HA HA! Uh-oh, that evil laughing looks kind of conspicuous in a doc file. I better put in some more technical info. If you want to write your own ncurses game, you can use the framework provided by robotfindskitten or the more flexible framework provided by the sequel (also on my software page). Now, let us part with the secret geek handshake!



## 7 Join the Bludgeoning robotfindskitten Community!

I want robotfindskitten to be the most comprehensive Zen simulation on the market. But for that to happen, I need your help. Please, send in any descriptions of objects you would like to see in robotfindskitten to [leonardr@ucla.edu](mailto:leonardr@ucla.edu). Heck, you could even tell me to put you in the program! I'm certainly not above that sort of thing. Pete Peterson set a great example by providing over 30 objects for everyone's kitten-finding enjoyment. Also, be sure to tell me how you like robotfindskitten.

In addition, as of November 2000, robotfindskitten has a project on Sourceforge! Simply visit <http://sourceforge.net/projects/rfk> and have a look at our bug tracking system, bickering flame war mailing lists, and haphazardly-branched CVS trees!



## Appendix A A Final Thought

Day and night I feverishly worked upon the machine, creating both a soul which could desire its goal, and a body with which it could realize it. Many who saw my creation called it an abomination, and denied me grant money. But they could not dissuade me from my impossible task. It was a spectre that tormented me always, a ghost I had to give a form and a life, lest it consume me from the inside. And when at last my task was done, when the grey box on wheels was complete and when it, as well as I, knew what had to be done, I felt deep sympathy for the machine. For I had not destroyed the phantom, but merely exorcized it into another body. The robot knew not why this task had to be performed, for I could not imbue it with knowledge I did not myself possess. And at the same time, I felt a sweeping sense of relief sweep over me, that somehow, the dream that had driven me for my entire life had come one step closer to fruition.

As I vocally activated the robot, I realized that it was following my instructions, but not out of any desire to obey me. Had I remained silent, it would have performed exactly the same operations. We were two beings controlled by the same force now. And yet, seeking vainly to hold some illusion of control over the machine I thought I had created, I gave my final command.

‘GO!’ I told the box as it began to roll out of my workshop into the frozen desert beyond. ‘FIND KITTEN!’

—The Book of Found Kittens, pages 43-4, author unknown



# Concept Index

## A

abomination . . . . . 15

## B

Bad People of the Future . . . . . 11  
 beta . . . . . 9  
 bickering . . . . . 13  
 bludgeon . . . . . 13  
 Book of Found Kittens . . . . . 15  
 Borland . . . . . 11

## C

clue-by-four . . . . . 13  
 community . . . . . 13  
 consistency . . . . . 7  
 contributing . . . . . 13  
 controls . . . . . 3  
 customized objects . . . . . 13

## D

Danger! . . . . . 5  
 destiny . . . . . 3  
 DOS . . . . . 11

## E

ending the simulation prematurely . . . . . 3

## F

find kitten . . . . . 15  
 foolish NSF grants . . . . . 15

## G

geek handshake . . . . . 11  
 Gort, Klaatu Verada Nikto . . . . . 15

## H

hippy commune . . . . . 13

## I

Independence Day . . . . . 11  
 Isn't it fun reading through the index? . . . . . 11

## K

kitten . . . . . 3

## M

movement . . . . . 3

## N

ncurses 5.0 . . . . . 11  
 Nerth Pork . . . . . 1  
 non-debian distribution of GNU/Linux . . . . . 11  
 non-kitten objects . . . . . 3

## O

original version . . . . . 11

## P

Peter Peterson II . . . . . 1, 13  
 Peterson, Pete . . . . . 1, 13

## R

riboflavin . . . . . 3, 11  
 robot . . . . . 3

## S

sheeplike masses . . . . . 11  
 simulation, Zen . . . . . 1  
 soul . . . . . 15  
 sourceforge . . . . . 13  
 spot kitten every time . . . . . 7  
 subversion . . . . . 3

## T

title screen . . . . . 3  
 touch . . . . . 3  
 Turbo C++ 3.0 . . . . . 11

**W**

warning ..... 5, 7, 9  
weird things ..... 9

**Y**

your tax dollars at work ..... 15

**Z**

Zen simulation ..... 1, 13

## Table of Contents

<b>1</b>	<b>The Niceties.....</b>	<b>1</b>
<b>2</b>	<b>The Documentation.....</b>	<b>3</b>
2.1	The Parts of the Simulation.....	3
2.2	Controls.....	3
<b>3</b>	<b>A Word of Warning.....</b>	<b>5</b>
<b>4</b>	<b>Another Word of Warning.....</b>	<b>7</b>
<b>5</b>	<b>Yet Another Word of Warning.....</b>	<b>9</b>
<b>6</b>	<b>Technical Information for Hard-Core Geeks</b>	
	.....	<b>11</b>
<b>7</b>	<b>Join the Bludgeoning robotfindskitten</b>	
	<b>Community!.....</b>	<b>13</b>
	<b>Appendix A A Final Thought .....</b>	<b>15</b>
	<b>Concept Index .....</b>	<b>17</b>