Robot Finds Kitten

Robot Finds Kitten is a program with a history. Originally written in DOS in 1997 for an online contest, RFK was in fact the only entry for the contest. RFK would have likely faded into obscurity had it not been re-written for Linux in 1999. Once available for Linux it's fame and fan base spread. You can read more about the programs history, it's fan base, and see other ports that are available at http://www.robotfindskitten.org.

RFK is a Zen simulation. There are no points to gain, nothing to achieve. You are a robot and you are trying to find a Kitten. There are many things around which are not kitten and apparently you are unable to distinguish visually between them so you are left to analyze several non-kitten objects before finding your goal. What really makes the game, tho, is the myriad of quirky descriptions of non-kitten things that you could run into. This version reads from an external file called messages.txt for it's descriptions, but if messages.txt is unavailable it will use a stock of 25 messages preloaded into the code, tho it's not the same.

Robot Finds Kitten is modified from Leonard Richardson's (leonardr@segfault.org) original code by Joseph Larson.

RFK.C	You will need: a C/C++ complier with curses support

```
#include <curses.h>
#include <time.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>

#define EMPTY -1
#define ROBOT 0
#define KITTEN 1
#define MAXMESSAGES 1000

void instructions();
void draw_robot();
void draw_kitten();
void process_input(int input, int** screen);
void play ();

typedef struct object
{
  int x;
  int y;
  int color;
  int character;
} OBJECT;

OBJECT robot;
OBJECT kitten;
OBJECT empty;
OBJECT bogus[MAXMESSAGES];
int bogus_messages[MAXMESSAGES];
char messages[MAXMESSAGES][81] =
{
  "This is not a kitten.",
  "That's just an old tin can.",
  "It's an altar to the horse god.",
  "A mere collection of pixels.",
  "A box of fumigation pellets.",
  "More grist for the mill.",
  "It's a square.",
  "Run away!  Run away!",
  "The rothe hits!  The rothe hits!",
  "This place is called Antarctica. There is no kitten here.",
  "It's a copy of \"Zen and The Art of Robot Maintenance\".",
  "\"Yes!\" says the bit.",
  "\"No!\" says the bit.",
  "A robot comedian. You feel amused.",
  "A forgotten telephone switchboard.",
  "It's a desperate' plug for Cymon's Games, http://www.cymonsgames.com",
  "The letters O and R.",
  "\"Blup, blup, blup\" says the mud pot.",
  "Another rabbit? That's three today!",
  "Thar's Mobius Dick, the convoluted whale. Arrr!",
  "This object here appears to be Louis Farrakhan's bow tie.",
  "...thingy???",
  "Pumpkin pie spice.",
  "Chewing gum and baling wire.",
  "It's the crusty exoskeleton of an arthropod!",
};
int used_messages[MAXMESSAGES];
int numbogus;
int nummessages;

int main(int argc, char *argv[])
{
  int supplied_number; /*Number of bogus OBJECTs, supplied on command
line. */
  int c, c2;
  int** screen;
  FILE *fp;
```

RFK.C	Listing continued from previous column...

```
nummessages = 0;
fp = fopen ("messages.txt", "r");
if (fp == NULL) {
  nummessages = 25;
}
else
{
  while (fgets (messages[nummessages++], 81, fp)) {
    c = strlen(messages[nummessages - 1]);
    if (c > 1) messages[nummessages - 1][c - 1] = 0;
    else nummessages--;
  }
}

if (argc == 1)
{
 numbogus = 20;
} else {
 supplied_number = atoi(argv[1]);
 if (supplied_number < 0 || supplied_number > nummessages)
 {
  printw ("Error: run-time parameter must be between 0 and %d."
   , nummessages);
  endwin (); exit(0);
 } else {
  numbogus = supplied_number;
 }
}
/* Do general start-of-program stuff. */
srand(time(NULL));
initscr ();
raw (); noecho(); curs_set(0); nonl(); keypad(stdscr,1);
start_color();
init_pair(0, COLOR_WHITE & A_BOLD, 0); /* for text */
init_pair(1, COLOR_WHITE, 0); /* for robot */
for (c = 2; c < COLORS; c++) init_pair(c, c, COLOR_BLACK);

empty.x = -1;
empty.y = -1;
empty.color = 0;
empty.character = ' ';

screen = (int**)malloc((COLS + 1) * sizeof (int*));
for (c = 0; c <= COLS; c++)
  screen[c] = (int*)malloc((LINES + 1) * sizeof (int));

/* Now we have the filling in of the various arrays.
 * Create an array to represent the screen so that we can make sure all
 * the OBJECTs go on different squares. */
for (c = 0; c <= COLS; c++)
{
  for (c2 = 0; c2 <= LINES; c2++)
  {
    screen[c][c2] = EMPTY;
  }
}
/* Create an array to ensure we don't get duplicate messages. */
for (c = 0; c < nummessages; c++)
{
  used_messages[c] = 0;
  bogus_messages[c] = 0;
  bogus[c] = empty;
}

/* Now we initialize the various game OBJECTs.
 * Assign a position to the player. */
robot.x = rand() % COLS;
robot.y = rand() % (LINES - 3) + 3;
robot.character = '#';
robot.color = 1;
screen[robot.x][robot.y] = ROBOT;
```

```c
  /* Assign the kitten a unique position. */
  do
  {
    kitten.x = rand() % COLS;
    kitten.y = rand() % (LINES - 3) + 3;
  } while (screen[kitten.x][kitten.y] != EMPTY);

  /* Assign the kitten a character and a color. */
  do
  {
    kitten.character = rand() % 254 + 1;
  } while (kitten.character == '#' || kitten.character == ' ');
  kitten.color = rand() % (COLORS - 2) + 2;
  screen[kitten.x][kitten.y] = KITTEN;

  /* Now, initialize non-kitten OBJECTs. */
  for (c = 0; c < numbogus; c++)
  {
    /* Assign a unique position. */
    do
    {
      bogus[c].x = rand() % COLS;
      bogus[c].y = rand() % (LINES - 3) + 3;
    } while (screen[bogus[c].x][bogus[c].y] != EMPTY);
    screen[bogus[c].x][bogus[c].y] = c+2;

    /* Assign a character. */
    do
    {
      bogus[c].character = rand() % 254 + 1;
    } while (bogus[c].character == '#' || bogus[c].character == ' ');
    bogus[c].color = rand() % (COLORS - 2) + 2;;

    /* Assign a unique message. */
    int index = 0;
    do
    {
      index = rand() % nummessages;
    } while (used_messages[index] != 0);
    bogus_messages[c] = index;
    used_messages[index] = 1;
  }

  /* Print instructions. */
  clear();
  instructions();
  clear();

  /* Put stuff on the screen. */
  attrset (COLOR_PAIR(0));
  move(0,0);
  attron (A_BOLD);
  printw ("robotfindskitten v22July2008");
  move(2,0);
  for (c = 0; c < COLS; c++) addch ('_');
  attroff (A_BOLD);
  move(kitten.y,kitten.x);
  draw_kitten();

  move (3,0);
  for (c = 0; c < numbogus; c++)
  {
    move(bogus[c].y,bogus[c].x);
    attrset (COLOR_PAIR(bogus[c].color));
    addch(bogus[c].character|A_ALTCHARSET);
  }

  move(robot.y,robot.x);
  draw_robot();
  int old_x = robot.x;
  int old_y = robot.y;

  /* Now the fun begins. */
  int input;
  input = getch();
  while (tolower(input) != 'q')
  {
    process_input(input, screen);
    /* Redraw robot, where avaliable */
    if (!(old_x == robot.x && old_y == robot.y))
    {
      move(old_y,old_x);
      addch(' ');
      move(robot.y,robot.x);
      draw_robot();
      screen[old_x][old_y] = EMPTY;
      screen[robot.x][robot.y] = ROBOT;
      old_x = robot.x;
      old_y = robot.y;
    }
    input = getch();
  }
  endwin ();
  return 0;
}

/* Given the keyboard input, interprets it. */
void process_input(int input, int** screen)
```

```c
{
  int check_x = robot.x;
  int check_y = robot.y;
  int c;

  switch (input)
  {
    case KEY_UP: /* up */
      check_y--;
      break;
    case KEY_DOWN: /* down */
      check_y++;
      break;
    case KEY_LEFT: /* left */
      check_x--;
      break;
    case KEY_RIGHT: /* right */
      check_x++;
      break;
    case 0:
      break;
    default: /* invalid command */
      move(1,0);
      attrset (COLOR_PAIR(0));
      printw("Invalid command: Use direction keys or Press 'Q'.");
      clrtoeol ();
      return;
  }

  /* Check for going off the edge of the screen. */
  if (check_y < 3 || check_y > LINES - 1 || check_x < 0 || check_x > COLS
- 1)
  {
    return;
  }

  /* Check for collision */
  if (screen[check_x][check_y] != EMPTY)
  {
    switch (screen[check_x][check_y])
    {
      case ROBOT:
        /* We didn't move. */
        break;
      case KITTEN: /* Found it! */
        move(1,0); clrtoeol ();

        /* The grand cinema scene. */
        for (c = 0; c <=3; c++)
        {
          move(1, COLS / 2 - 5 + c);
          addch(' ');
          move(1,COLS / 2 + 4 - c);
          addch(' ');
          move(1,COLS / 2 - 4 + c);
          if (input == KEY_LEFT || input == KEY_UP)
            draw_kitten();
          else
            draw_robot();
          move(1,COLS / 2 + 3 - c);
          if (input == KEY_LEFT || input == KEY_UP)
            draw_robot();
          else
            draw_kitten();
          refresh (); napms (600);
        }
        /* They're in love! */
        move(0,COLS / 2 - 1);
        attrset (COLOR_PAIR(COLOR_RED)|A_BOLD);
        addch(3|A_ALTCHARSET);
        addch(3|A_ALTCHARSET);
        move(1,0);
        refresh(); napms (1200);
        attrset (COLOR_PAIR(0)); attron (A_BOLD);
        printw ("You found kitten! Way to go, robot!");
        attroff (A_BOLD);
        getch ();
        endwin (); exit(0);
        break;
      default:
        move(1,0);
        attron (A_BOLD);
        printw (messages[bogus_messages[screen[check_x][check_y]-2]]);
        clrtoeol ();
        attroff (A_BOLD);
        break;
    }
    move(2,0);
    return;
  }
  /* Otherwise, move the robot. */
  robot.x = check_x;
  robot.y = check_y;
}

void instructions()
{
  attrset (COLOR_PAIR(0)|A_BOLD);
  move(0,0);
  printw ("robotfindskitten v22July2008\n"
```

| RFK.C | Listing continued from page 2... | messages.txt |

```c
    "Originally by the illustrious Leonard Richardson 1997\n"
    "Written especially for the Nerth Pork robotfindskitten contest\n"
    "ReWritten in PDCurses by Joseph Larson\n"
    "Especially for Cymon's Games 2008 HTTP://WWW.CYMONSGAMES.COM\n\n"
    "   In this game, you are robot (");
  draw_robot();
  attrset (COLOR_PAIR(0)|A_BOLD);
  printw ("). Your job is to find kitten. This task\n"
  "is complicated by the existance of various things which are not\n"
  "kitten. Robot must touch items to determine if they are kitten or\n"
  "not. The game ends when robotfindskitten. Alternatively, you may\n"
  "end the game by hitting the 'Q' key.\n\n"
  "   Press any key to start.\n");
  getch();

}

void draw_robot() /* Draws robot at current position */

{
  attrset (COLOR_PAIR(robot.color));
  addch(robot.character);
}

void draw_kitten() /* Draws kitten at current position */

{
  attrset (COLOR_PAIR(kitten.color));
  addch(kitten.character|A_ALTCHARSET);
}
```

messages.txt

```
"I pity the fool who mistakes me for kitten!" sez Mr. T.
That's just an old tin can.
It's an altar to the horse god.
A box of dancing mechanical pencils. They dance! They sing!
It's an old Duke Ellington record.
A box of fumigation pellets.
A digital clock. It's stuck at 2:17 PM.
That's just a charred human corpse.
I don't know what that is, but it's not kitten.
An empty shopping bag. Paper or plastic?
Could it be... a big ugly bowling trophy?
A coat hanger hovers in thin air. Odd.
Not kitten, just a packet of Kool-Aid(tm).
A freshly-baked pumpkin pie.
A lone, forgotten comma, sits here, sobbing.
ONE HUNDRED THOUSAND CARPET FIBERS!!!!!
It's Richard Nixon's nose!
It's Lucy Ricardo. "Aaaah, Ricky!" she says.
You stumble upon Bill Gates' stand-up act.
Just an autographed copy of the Kama Sutra.
It's the Will Rogers Highway. Who was Will Rogers, anyway?
Leonard Richardson is here, asking people to lick him.
It's a stupid mask, fashioned after a beagle.
Your State Farm Insurance(tm) representative!
It's the local draft board.
Seven 1/4" screws and a piece of plastic.
An 80286 machine.
One of those stupid "Homes of the Stars" maps.
A signpost saying "TO KITTEN". It points in no particular direction.
A hammock stretched between a tree and a volleyball pole.
A Texas Instruments of Destruction calculator.
It's a dark, amphorous blob of matter.
Just a pincushion.
It's a mighty zombie talking about some love and prosperity.
"Dear robot, you may have already won our 10 MILLION DOLLAR prize..."
It's just an object.
A mere collection of pixels.
A badly dented high-hat cymbal lies on its side here.
A plush Chewbacca.
Daily hunger conditioner from Australasia
Just some stuff.
Why are you touching this when you should be finding kitten?
A glorious fan of peacock feathers.
It's some compromising photos of Babar the Elephant.
A copy of the Weekly World News. Watch out for the chambered nautilus!
It's the proverbial wet blanket.
A "Get Out of Jail Free" card.
An incredibly expensive "Mad About You" collector plate.
```

| messages.txt | |

```
Paul Moyer's necktie.
A haircut and a real job. Now you know where to get one!
An automated robot-hater. It frowns disapprovingly at you.
An automated robot-liker. It smiles at you.
It's a black hole. Don't fall in!
Just a big brick wall.
You found kitten! No, just kidding.
Heart of Darkness brand pistachio nuts.
A smoking branding iron shaped like a 24-pin connector.
It's a Java applet.
An abandoned used-car lot.
A shameless plug for Crummy: http://www.crummy.com/
A shameless plug for the UCLA Linux Users Group: http://linux.ucla.edu/
It's a desperate' plug for Cymon's Games, http://www.cymonsgames.com/
A can of Spam Lite.
This is another fine mess you've gotten us into, Stanley.
It's scenery for "Waiting for Godot".
This grain elevator towers high above you.
A Mentos wrapper.
It's the constellation Pisces.
It's a fly on the wall. Hi, fly!
This kind of looks like kitten, but it's not.
It's a banana! Oh, joy!
A helicopter has crashed here.
Carlos Tarango stands here, doing his best impression of Pat Smear.
A patch of mushrooms grows here.
A patch of grape jelly grows here.
A spindle, and a grindle, and a bucka-wacka-woom!
A geyser sprays water high into the air.
A toenail? What good is a toenail?
You've found the fish! Not that it does you much good in this game.
A Buttertonsils bar.
One of the few remaining discoes.
Ah, the uniform of a Revolutionary-era minuteman.
A punch bowl, filled with punch and lemon slices.
It's nothing but a G-thang, baby.
IT'S ALIVE! AH HA HA HA HA!
This was no boating accident!
Wait! This isn't the poker chip! You've been tricked! DAMN YOU, MENDEZ!
A livery stable! Get your livery!
It's a perpetual immobility machine.
"On this spot in 1962, Henry Winkler was sick."
There's nothing here; it's just an optical illusion.
The World's Biggest Motzah Ball!
This appears to be a rather large stack of trashy romance novels.
Look out! Exclamation points!
A herd of wild coffee mugs slumbers here.
It's a limbo bar! How low can you go?
It's the horizon. Now THAT'S weird.
A vase full of artificial flowers is stuck to the floor here.
A large snake bars your way.
A pair of saloon-style doors swing slowly back and forth here.
It's an ordinary bust of Beethoven... but why is it painted green?
It's TV's lovable wisecracking Crow! "Bite me!" he says.
Hey, look, it's war. What is it good for? Absolutely nothing
It's the amazing self-referential thing that's not kitten.
A flamboyant feather boa. Now you can dress up like Carol Channing!
"Sure hope we get some rain soon," says Farmer Joe.
"Topsoil's all gone, ma," weeps Lil' Greg.
A team of arctic explorers is camped here.
This object here appears to be Louis Farrakhan's bow tie.
This is the world-famous Chain of Jockstraps.
A trash compactor, compacting away.
This toaster strudel is riddled with bullet holes!
It's a hologram of a crashed helicopter.
This bologna has a first name, it's R-A-N-C-I-D.
A salmon hatchery? Look again. It's merely a single salmon.
It's a rim shot. Ba-da-boom!
This is an anagram.
This object is like an analogy.
It's a symbol. You see in it a model for all symbols everywhere.
The object pushes back at you.
A traffic signal. It appears to have been recently vandalized.
This is a Lagrange point. Don't come too close now.
The dirty old tramp bemoans the loss of his harmonica.
Look, it's Fanny the Irishman!
What in blazes is this?
It's the instruction manual for a previous version of this game.
A brain cell. Oddly enough, it seems to be functioning.
Tea and/or crumpets.
This jukebox has nothing but Cliff Richards albums in it.
It's a Quaker Oatmeal tube, converted into a drum.
```
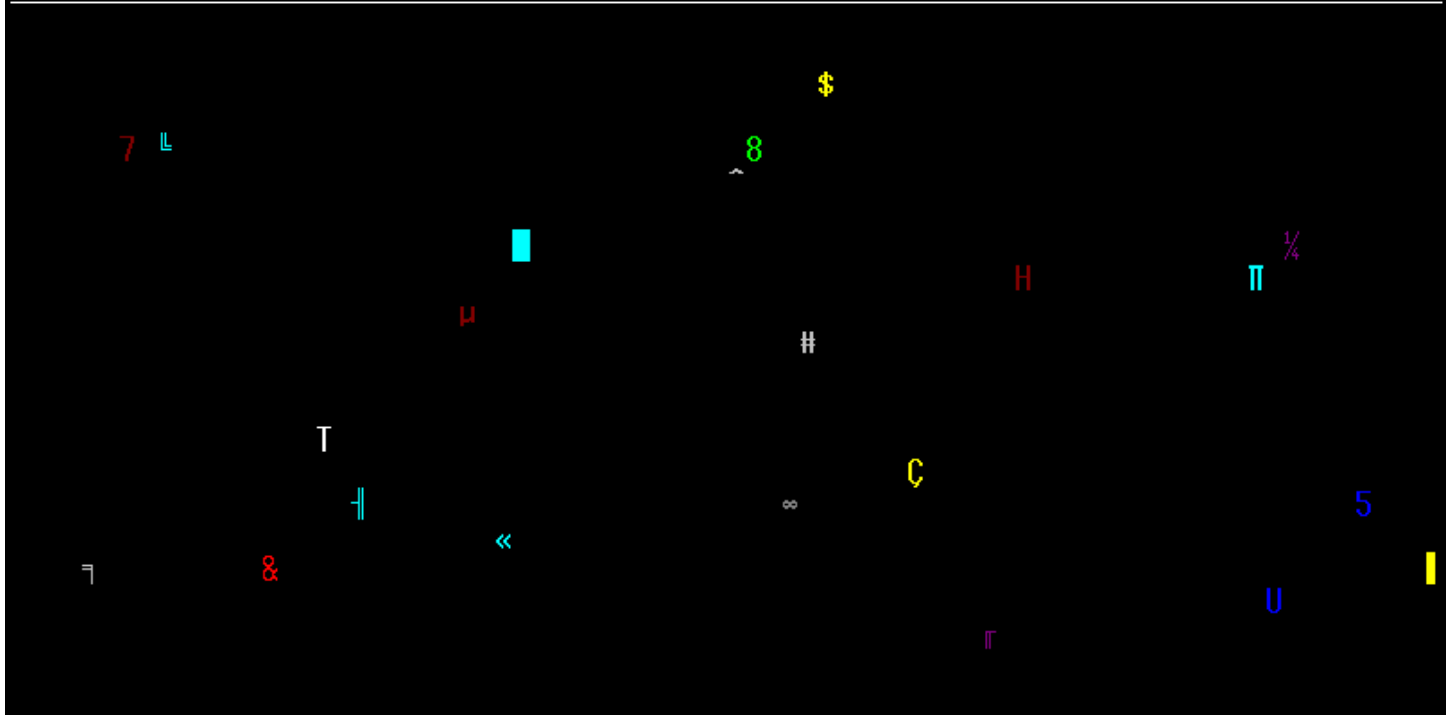
Author's Notes:

The messages.txt presented here is still only a small sample of the whole listing that can be found online.

While a curses version of Robot Finds Kitten did already exist I was having difficulty getting it to run with PDCurses, so I decided to do my own. The conversion went remarkably smoothly. This version differs from the original code in that:

- PDCurses is a C library, so I've changed the source to C (not a big deal because most of the code except the output didn't use any C++ specific functionality, and the output got changed to PDCurses)
- This version loads descriptions from messages.txt if it finds one.
- Consistant with Curses, this version will play with any screen size, and that was a bit of an educational experience.
- I've added my own shameless plug to the messages.txt and the source

Aside from these minor changes the code is still true in all other way to the original, right down to the overly large main function and formatting preferences.

```
   [-]
   (+)=C
   | |
   OOO
```

```
robotfindskitten v22July2008
Another rabbit? That's three today!
```

```
                                         $
      7 ⌞                          8
                                  ~
                      ■
              μ                              ¼
                                    H      π
                          #
         T
           ╢                    Ç
                       «            ∞              5
    ⌐       &                              U       ▌
                              ╓
```

```
   [-]
   (+)=C
   | |
   OOO
```