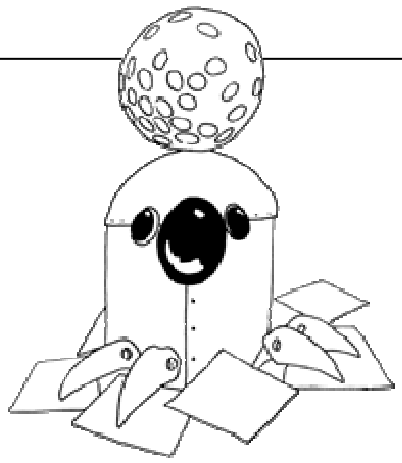
Putt

Is there anything more relaxing than a day on the green? What if the green could come to you.

Putt is a golf simulator… sort of. In this game all you have to do is input how hard to hit the ball and the game plays itself. If you hit the ball 1 unit you'll only go 1 square in distance, but if that distance is in a diagonal direction you'll need to adjust your strength. Numbers nerds at this point would pull out Pythagorean theorem. Everyone else will just want to use trail and error. Every other square on the green is marked with a dot to help you count, if you are so inclined.

When you land your ball in the hole it will seal itself up and a new hole will appear. After 18 holes your score it tallied.

Putt is by Joseph Larson.



/* putt.c listing begins: */

```
#include <iostream>
#include <fstream>
#include <cmath>
#include <cstdlib>
#include <ctime>
using namespace std;

#define MAXX 79
#define MAXY 23
#define PAR 2
#define NUMHOLES 18

char holegfx[3][3] =
{{'/','|','\\'},{'-',' ','-'},{'\\','|','/'}};
int ballx, bally, holex, holey, anim = 1;

int dist () {
  double dx, dy;

  dx = (holex - ballx);
  dy = (holey - bally);
  return (int)(sqrt (dx * dx + dy * dy));
}

void drawgreen (void) {
  int x, y;

  for (y = 0; y < MAXY; y++) {
    for (x = 0; x < MAXX; x++) {
      if (x == ballx && y == bally) cout << 'o';
      else if ((abs (x - holex) < 2) && (abs (y - holey) < 2))
        cout << holegfx[1 + x - holex][1 + y - holey];
      else cout << ". "[(x+y)%2];
    }
    cout << '\n';
  }
}

void wait (int ms) {
  int start, current;

  start = clock();
  do {
    current = clock();
  } while (current - start < ms);
}

int shootball (double force) {
  double dx, dy, sf;
  int finalx, finaly, ret = 0;

  dx = holex - ballx;
  dy = holey - bally;
  if (force > dist ()) { // overshoot the hole?
    force -= (force - dist ()) / 2; ret = 1;
  }
  sf = force / dist ();
  dx *= sf; dy *= sf;
  finalx = ballx + (int)round (dx); finaly = bally + (int)round (dy);
  if ((finalx < 0) || (finaly < 0) || (finalx >= MAXX) || (finaly >= MAXY)) {
    finalx -= (int)round (dx); finaly -= (int)round (dy); ret = 2;
```

/* Listing continued on next page…*/

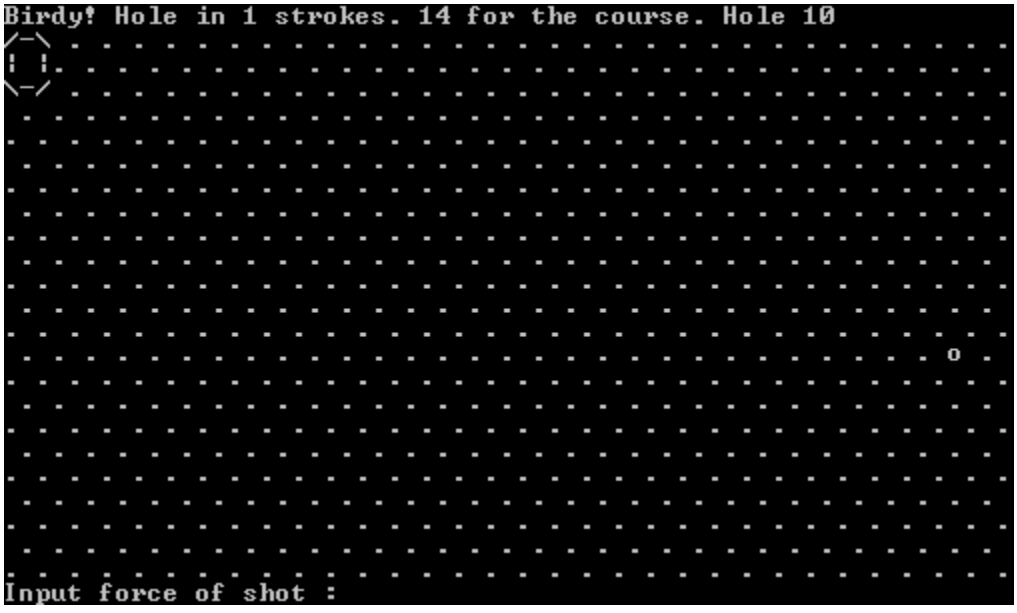
/* Listing continued from previous page */

```
  }
  if (anim) do {
    dx /= 2; dy /= 2;
    ballx += (int)round(dx); bally += (int)round(dy);
    drawgreen ();
    wait (150);
    cout << '\n';
  } while (!((abs(dx) <= 1.0 ) && (abs(dy) <= 1.0))
    && (ballx >= 0) && (bally >= 0) && (ballx < MAXX) && (bally < MAXY));
  ballx = finalx; bally = finaly;
  return ret;
}

void playgame (void){
  int shot, hole, tstroke = 0;
  int hstroke[NUMHOLES];

  ballx = rand () % MAXX; bally = rand () % MAXY;
  for (hole = 0; hole < NUMHOLES; hole++) {
    do { holex = rand () % MAXX; holey = rand () % MAXY;
    } while (dist () < 4);
    hstroke[hole] = 0;
    while (dist() > 1) {
      hstroke[hole] ++; tstroke ++;
      cout << "Hole " << (hole + 1) << '\n';
      drawgreen ();
      cout << "Input force of shot : ";
      cin >> shot;;
      switch (shootball(shot)) {
        case 1 : cout << "Over Shot the Hole. "; break;
        case 2 : cout << "Off the Edge. Return to start. "; break;
      }
    }
    cout << '\n';
    if (hstroke[hole] == PAR) cout << "Par! ";
    if (hstroke[hole] < PAR) cout << "Birdy! ";
    cout << "Hole in " << hstroke[hole] << " strokes. "
      << tstroke << " for the course. ";
  }
  cout << "\n\nScore Card:\nTotal Strokes : " << tstroke;
  if (tstroke == (PAR * NUMHOLES)) cout << ", Par for the course\n";
  else if (tstroke <= (PAR * NUMHOLES))
    cout << ", " << (PAR * NUMHOLES - tstroke) << " under par!\n";
  else cout << '\n';
  cout << "Hole\tStrokes\n";
  for (hole = 0; hole < NUMHOLES; hole++)
    cout << (hole + 1) << '\t' << hstroke[hole] << '\n';
}

int playagain (void) {
  char input;

  cout << "\nDo you want to play again? (y\\n) ";
  cin >> input;
  if (toupper(input) == 'Y') return 1;
  else return 0;
}

int main (void) {
  char input;

  cout << "Putt\n----\n"
  << "A relaxing day on the green is only moments away. When you see the\n"
```

/* Listing continued on next page…*/

/* Listing continued from previous page */

```
  << "green input how hard to hit the ball (the distance to the hole). Aim\n"
  << "will be handled for you.\n"
  << "If you overshoot the ball will slow down. If you shoot over the edge\n"
  << "of the screen you lose a stroke and shoot from where you were.\n"
  << "Good luck.\n\n"
  << "(Hint: The green itself is " << MAXX << "x" << MAXY << ".)\n\n"
  << "Do you want to see the shots animated? (y/n) ";
  cin >> input;
  if (toupper (input) == 'N') anim = 0; else anim = 1;
  srand (time (NULL));
  do playgame (); while (playagain ());
  cout << "Thank you for playing\n";
}
```



/* Listing continued on next page…*/