Rock Paper Scissors

It's the ancient combat triangle: rock smashes scissors, scissors cut paper, paper covers rock. Countless decisions have rested on mastery of this tenuous balance.

So to prepare for those future decisions this program allows you to practice against a computer opponent. 5 computer opponents in fact. Win all 5 rounds an you can declare yourself the champion. Each of the 5 rounds the opponents act differently. Knowing how they act could give you an edge. Maybe.

If the standard "rock/paper/scissors" combination is too tame for you all you need to do is edit the rps.txt file that the game reads for it's pictures and make your own challenges. Make sure the data file is 18 lines, each 6 line block starts with the name of the symbol, the next 5 lines are the symbol, no more than 12 characters wide. Then the next 6 lines will be the symbol that beats it.

RPS was made by James E. Ward.

/* rps.c listing begins: */

```
#include <cstdlib>
#include <curses.h>
#include <time.h>
#include <string>
#include <fstream>
#define LIFE 5

using namespace std;

char pchoice='0';
int cchoice=0,plife=LIFE,clife=LIFE,level=1,pref=1,winner=3;
string lines[18];

void init () {
  int c=0;
  srand (time(NULL));
  keypad(initscr(),1);
  cbreak();
  noecho();
  curs_set(0);
  nonl();
  /*Load output*/
  ifstream yup("rps.txt");
  for(int i=0;i<18;i++)
  {
      string line;
      getline (yup, line);
      lines[i]=line;
  }
  clear();
  mvprintw(0,0,"Welcome to \nBattle");
  mvprintw(2,0,lines[0].c_str());
  mvprintw(3,0,lines[6].c_str());
  mvprintw(4,0,lines[12].c_str());
  getch();
}

void draw()
{
  clear();
  int pline=(pchoice=='1')?0:(pchoice=='2')?1:2;
  int cline=cchoice-1;
  mvprintw(0,0,"Your Life: %d  Enemy #%d Life: %d",plife,level,clife);
  mvprintw(1,0,"Your choice:       Enemy choice:");
  if(cchoice!=0){
  for(int i=0;i<6;i++)
  {
    mvprintw(3+i,5,lines[i+(6*pline)].c_str());
    mvprintw(3+i,18,lines[i+(6*cline)].c_str());

  }}
  switch (winner)
  {
    case 0:mvprintw(10,0,"Tie");break;
    case 1:mvprintw(10,0,"You win");break;
    case 2:mvprintw(10,0,"You lose");break;
    default:/*mvprintw(10,0,"ERROR");*/break;
  }
  mvprintw(12,0,"1:");
  mvprintw(12,2,lines[0].c_str());
  mvprintw(13,0,"2:");
```

/* Listing continued on next page…*/

/* Listing continued from previous page */

```
  mvprintw(13,2,lines[6].c_str());
  mvprintw(14,0,"3:");
  mvprintw(14,2,lines[12].c_str());
}

int ai()
{
  int a;
  switch(level)
  {
    case 1: a=rand()%4+1;return(a==4)?pref:a;
    case 2: a=rand()%4+1;return(a==4)?pref:a;
    case 3: a=rand()%4+1;return(a==4)?pref:a;
    case 4: return rand()%3+1;
    case 5: a=rand()%4+1;
      if (a!=4){return a;}
      else {return(cchoice=='1')?2:(cchoice=='2')?3:1;}
    default:return rand()%3+1;
  }
}

int rps() /*returns 1 for player 2 for computer or 0 for draw*/
{
  switch(pchoice)
  {
    case '1':return(cchoice==2)?2:(cchoice==3)?1:0;
    case '2':return(cchoice==3)?2:(cchoice==1)?1:0;
    case '3':return(cchoice==1)?2:(cchoice==2)?1:0;
    default:return 2;
  }
}

int main()
{
  init();
  pref=rand()%3+1;
  draw();
  while(plife>0)
  {
    while(1)
    {
      pchoice=getch();
      if (pchoice=='c'||pchoice=='C'||pchoice=='x'||pchoice=='x')
        exit(endwin());
      if(pchoice=='1'||pchoice=='2'||pchoice=='3')break;
    }
    cchoice=ai();
    if ((winner=rps())==2){--plife;}
    if (winner==1)
    {
      --clife;
      if (clife<1)
      {
        clear();
        ++level;
        if (level>5)
        {
          mvprintw(0,0,"Congratulations you win!!");
          getch();
          exit(endwin());
        }
        cchoice=0;
        winner=0;
```

/* Listing continued on next page…*/

/* Listing continued from previous page */

```
        mvprintw(0,0,"Next level");
        getch();
        plife=LIFE;
        clife=LIFE;
        winner=3;
        pref=rand()%3+1;
        draw();
      }
    }
    draw();
  }
  clear();
  mvprintw(0,0,"You lost to opponent #%d with %d lives left.",level,clife);
  getch();
  mvprintw(1,0,"Thank You for playing");
  getch();
  exit(endwin());
}
```

Choose one of the following:

rps.txt

```
Rock
/===\
\===/
 >-<
/===\
\===/
Paper
+---+
|pa |
| p |
| er|
+---+
Scissors
\   /
 \ /
  X
 / \
0   0
```

rps.txt

```
Knife
   |
   |
   +
   |

Sword
   ^
  |||
  |||
o---o
   |
Gun
'=====
{Q
||
||
```

rps.txt

```
Fire

     ^
    ^v^
   ^v^v^

Water

~~~~~~~~
~~~~~~~~
~~~~~~~~

Wood
   ____
  /-\~~\
((O))~~)
 \_/~~/
```

rps.txt

```
Rock
,-,___
  |___)
| |___)
 U____)
\(___)
Paper
/    _)___
     _____)
     ______)
     _____)
\________)
Scissors
,-,_,--"`)
  | ,=="_
| |____  )
 U____)"'
\(___)
```