Welcome to pig!

You will be playing a game to 100, against the computer.

It is your turn to roll again! (r to roll)...r

You rolled a 3 and 2.

You add 5 to your round total, which is now 5.

Do you wish to roll again or stop? (r to roll, s to stop): r

You rolled a 1 and 2.

Your rolled a 1! No points for this round.

Your total game score is 0.

The computer rolls a 1 and 4.

It rolled a 1! It doesn't get points this round.

Computer's total game score is 0.

It is your turn to roll again! (r to roll)...r

You rolled a 2 and 2.

You add 4 to your round total, which is now 4.

Do you wish to roll again or stop? (r to roll, s to stop): s

Your total game score is 4.

The computer rolls a 3 and 5.

Its current round total is now 8.

The computer rolls a 2 and 3.

Its current round total is now 13.

The computer rolls a 6 and 4.

Its current round total is now 23.

Computer's total game score is 23.

It is your turn to roll again! (r to roll)...r

You rolled a 5 and 1.

Your rolled a 1! No points for this round.

Your total game score is 4.

The computer rolls a 4 and 2.

Its current round total is now 6.

The computer rolls a 4 and 4.

Its current round total is now 14.

The computer rolls a 4 and 5.

Its current round total is now 23.

Computer's total game score is 46.

It is your turn to roll again! (r to roll)...r

You rolled a 3 and 1.

Your rolled a 1! No points for this round.

Your total game score is 4.

The computer rolls a 6 and 6.

Its current round total is now 12.

The computer rolls a 1 and 1.

It rolled double 1!

Computer's total game score is 0.

It is your turn to roll again! (r to roll)...r

You rolled a 2 and 6.

You add 8 to your round total, which is now 8.

Do you wish to roll again or stop? (r to roll, s to stop): r

You rolled a 6 and 2.

You add 8 to your round total, which is now 16.

Do you wish to roll again or stop? (r to roll, s to stop): r

You rolled a 1 and 2.

Your rolled a 1! No points for this round.

Your total game score is 4.

The computer rolls a 4 and 3.

Its current round total is now 7.

The computer rolls a 5 and 4.

Its current round total is now 16.

The computer rolls a 3 and 5.

Its current round total is now 24.

Computer's total game score is 24.

It is your turn to roll again! (r to roll)...r

You rolled a 2 and 3.

You add 5 to your round total, which is now 5.

Do you wish to roll again or stop? (r to roll, s to stop): r

You rolled a 4 and 2.

You add 6 to your round total, which is now 11.

Do you wish to roll again or stop? (r to roll, s to stop): r

You rolled a 5 and 1.

Your rolled a 1! No points for this round.

Your total game score is 4.

The computer rolls a 2 and 6.

Its current round total is now 8.

The computer rolls a 4 and 3.

Its current round total is now 15.

The computer rolls a 6 and 1.

It rolled a 1! It doesn't get points this round.

Computer's total game score is 24.

It is your turn to roll again! (r to roll)...r

You rolled a 1 and 3.

Your rolled a 1! No points for this round.

Your total game score is 4.

The computer rolls a 5 and 6.

Its current round total is now 11.

The computer rolls a 6 and 3.

Its current round total is now 20.

Computer's total game score is 44.

It is your turn to roll again! (r to roll)...r

You rolled a 4 and 1.

Your rolled a 1! No points for this round.

Your total game score is 4.

The computer rolls a 2 and 1.

It rolled a 1! It doesn't get points this round.

Computer's total game score is 44.

It is your turn to roll again! (r to roll)...r

You rolled a 5 and 6.

You add 11 to your round total, which is now 11.

Do you wish to roll again or stop? (r to roll, s to stop): r

You rolled a 4 and 5.

You add 9 to your round total, which is now 20.

Do you wish to roll again or stop? (r to roll, s to stop): s

Your total game score is 24.

The computer rolls a 4 and 6.

Its current round total is now 10.

The computer rolls a 3 and 2.

Its current round total is now 15.

The computer rolls a 2 and 5.

Its current round total is now 22.

Computer's total game score is 66.

It is your turn to roll again! (r to roll)...r

You rolled a 4 and 3.

You add 7 to your round total, which is now 7.

Do you wish to roll again or stop? (r to roll, s to stop): r

You rolled a 1 and 6.

Your rolled a 1! No points for this round.

Your total game score is 24.

The computer rolls a 4 and 2.

Its current round total is now 6.

The computer rolls a 2 and 1.

It rolled a 1! It doesn't get points this round.

Computer's total game score is 66.

It is your turn to roll again! (r to roll)...r

You rolled a 1 and 3.

Your rolled a 1! No points for this round.

Your total game score is 24.

The computer rolls a 6 and 5.

Its current round total is now 11.

The computer rolls a 4 and 2.

Its current round total is now 17.

The computer rolls a 4 and 6.

Its current round total is now 27.

Computer's total game score is 93.

It is your turn to roll again! (r to roll)...r

You rolled a 6 and 3.

You add 9 to your round total, which is now 9.

Do you wish to roll again or stop? (r to roll, s to stop): r

You rolled a 1 and 3.

Your rolled a 1! No points for this round.

Your total game score is 24.

The computer rolls a 5 and 2.

Its current round total is now 7.

Computer's total game score is 100.

YOU LOSE!!

Press any key to continue . . .

/\* Name: MP2

\* Date: 10 - 24 - 13

\* Author: Jason Duffey

\*

\*Plays a player vs. computer game of pig, where a pair of dice is rolled.

\*if a 1 shows up you lose all the points for the round, double 1 loses all

\*points in a game, if not the sum of the two dice is added to the round score

\*and the player has a choice of rolling again or stopping. First to 100 wins.

\*/

#include <iostream>

#include <ctime>

#include <cstdlib>

using namespace std;

const int WIN\_VALUE = 100; //value that the player and computer play to

int rollDie (void);

bool isTurnScoreLost (int die1value, int die2value);

bool isGameScoreLost (int die1value, int die2value);

char getUserInput (void);

void playerRound (int& score);

void cpuRound (int& score);

void checkWin (int player);

/\* \* \* \* \* MAIN \* \* \* \* \*/

int main ()

{

int cpuScore = 0; //keeps track of the computer's score

int playerScore = 0; //keeps track of the player's score

srand (time ((unsigned) 0));

cout << "Welcome to pig!" << endl

<< "You will be playing a game to 100, against the computer.\n\n";

//loops player rounds, and computer rounds until there is a winner

while (cpuScore < WIN\_VALUE && playerScore < WIN\_VALUE)

{

playerRound (playerScore);

cout << "Your total game score is " << playerScore << ".\n\n";

if (playerScore >= WIN\_VALUE)

{

break;

}

cpuRound (cpuScore);

cout << "Computer's total game score is " << cpuScore << ".\n\n";

}

//Determines who won the game

checkWin (playerScore);

return 0;

}

/\* \* \* \* \* FUNCTION DEFINITIONS \* \* \* \* \*/

/\*

\* int cpuRound: plays a round for the computer stops when computer has 20 points

\* Parameters: int score - passes the computers current game score

\* Return: the computers game score after the round

\*/

void cpuRound (int& score)

{

const int ROUND\_STOP = 20; //the minimum value on which the computer will stop rolling

int die1;

int die2;

int diceSum; //sum of the two dice on the current roll

int roundScore = 0; //current points for the round

//rolls the dice, checks for one or double one, then looks if it should roll again

// (if the round sum is less then 20

do

{

die1 = rollDie ();

die2 = rollDie ();

diceSum = die1 + die2;

cout << "\n The computer rolls a " << die1 << " and " << die2 << ".\n";

//checks for 1 or double 1

if ( isGameScoreLost(die1, die2) )

{

score = 0;

cout << "It rolled double 1!\n";

return;

}

if ( isTurnScoreLost(die1, die2) )

{

roundScore = 0;

cout << "It rolled a 1! It doesn't get points this round.\n";

break;

}

//adds the sum of the dice to the current

roundScore += diceSum;

cout << "Its current round total is now "

<< roundScore << ".\n";

if (score + roundScore >= WIN\_VALUE)

{

break;

}

}

while ( roundScore < ROUND\_STOP );

score += roundScore;

}

/\*

\* playerRound: plays through one round for the player

\* Parameters: int score - passes the players current game score

\* Return : players score for the round

\*/

void playerRound (int& score)

{

int die1;

int die2;

int diceSum; //sum of the two dice on the current roll

int roundScore = 0; //current points for the round

char roundStart; //stores the user's choice to roll the dice the first time

//asks the user to roll the die for their first roll

cout << endl << "It is your turn to roll again! (r to roll)...";

cin >> roundStart;

//rolls die, checks for 1 or double 1, adds points to round total, and

// repeats until the user tells it to stop

do

{

die1 = rollDie ();

die2 = rollDie ();

diceSum = die1 + die2;

cout << "\n You rolled a " << die1 << " and " << die2 << ".\n";

//checks for 1 or double one

if ( isGameScoreLost(die1, die2) )

{

score = 0;

cout << "You rolled double 1!\n";

return;

}

if ( isTurnScoreLost(die1, die2) )

{

roundScore = 0;

cout << "Your rolled a 1! No points for this round.\n";

break;

}

//if a 1 is not rolled

roundScore += diceSum;

cout << "You add " << diceSum << " to your round total, which is now "

<< roundScore << ".\n";

}

while ( getUserInput () == 'r');

score += roundScore;

}

/\*

\* rollDie: simulates the roll of a standard (6-sided) die.

\* Parameters: none

\* Returns: integer pip value (from 1 to 6) of the rolled die.

\*/

int rollDie (void)

{

const int NUMBER\_OF\_DIE\_SIDES = 6; //on a normal die

const int LOWEST\_DIE\_VALUE = 1;

return rand() % NUMBER\_OF\_DIE\_SIDES + LOWEST\_DIE\_VALUE;

}

/\*

\* getUserInput: gets the input from the user on whether or not to roll again

\* Parameters: none

\* Returns: r or s depending on wheter the user wants to roll(r) or stop(s)

\*/

char getUserInput (void)

{

char playerChoice;

//repeats prompt until a correct input is entered

do

{

cout << "Do you wish to roll again or stop? (r to roll, s to stop): ";

cin >> playerChoice;

}

while (playerChoice != 'r' && playerChoice != 'R' && playerChoice != 's' &&

playerChoice != 'S');

return tolower( playerChoice );

}

/\*

\* isTurnScoreLost: evaluates if the player has rolled a 1

\* Parameters: int die1value, int die2value - values of the two dice rolled

\* Return: true if a 1 has been rolled, false if not

\*/

bool isTurnScoreLost ( int die1value, int die2value )

{

if (die1value == 1 || die2value == 1)

{

return true;

}

return false;

}

/\*

\* isGameScoreLost: evaluates if the player has rolled double 1

\* Parameters: int die1value, int die2value - values of the two dice rolled

\* Return: true if double 1 has been rolled, false if not

\*/

bool isGameScoreLost ( int die1value, int die2value )

{

if (die1value == 1 && die2value == 1)

{

return true;

}

return false;

}

/\*

\* checkWin: checks to see who the winner is

\* Parameters: int player, the players score to see if they won, if not it is the computer

that won

\* Return: none

\*/

void checkWin (int player)

{

if (player >= WIN\_VALUE)

{

cout << "\n YOU WIN!! \n";

}

else

{

cout << "\n YOU LOSE!! \n";

}

}