Project 1

<Guessing Password>

Name: Haolan Ye (Benjamin)

Class: CSC-5 40717

Date: 01/31/2015

1. Inspiration:

This game is to guessing a 4-digit password. The first inspiration of making this game is from Hangman (A game of guessing words). The second inspiration is from a 4-digit coded lock of my luggage. After I travelled to a city, I forgot the password of my lock. Then, I tried lots of times to guess the pass-word. Every time after I guessed, I need to try to unlock it.

1. Introduction:

Different from Hangman, players should guess the numbers instead of words in Guessing Password. First, computer will generate a 4-digit number (from 0000 to 9999). Then, players only can guess one digit of password at one time. Players should input number they guess and the digit of this number in order. Afterward, the computer will tell you whether the number and digit are wrong or not. Players only have 12 chances to guess the password. If players cannot guess all digits of password after 12 chances, computer will display game over.

When players do not know how to play this game, they can type “s” and then press Enter to ask computer to display the sample input of this game. Players cannot guess the same number in a specific digit after this digit has been done.

Code:

/\*

\* File: main.cpp

\* Author: Haolan Ye(Benjamin)

\* Created on January 27, 2015, 12:32 PM

\* Purpose: Project1(Name of the game:Bomb Password)

\*/

//system Libraries

#include <iostream>

#include <cstdlib> //for random number

#include <string>

#include <vector>

#include <fstream> //file I/O

using namespace std;

//User Libraries

//Global Constants

//Function Prototypes

string toDash(int);//change the password to dash

void introduce();//introduce the game

void ask(char&,int&);//ask user for guessing

char check(char,int,const char[],int);//check whether number and digit are correct

bool indexOf(char,const char[],int);//return whether the char is in the char array

void replace(string& dash,char guess,int digit);//replace of the correct digit

bool inside(const vector<int>,int);//return whether this digit is finished

void sample();//display the sample of guessing

//Execution begins here

int main(int argc, char\*\* argv) {

//set seed for random number

srand(static\_cast<unsigned short>(time(0)));

//declare a file object

ofstream output;

//open the file

output.open("Times.dat");

//declare and initialize variables

const int TOTCHNS=15;//total chance of the game

const int SIZE=4;//the size of the char array

string dash;

string answer;

int times=0; //how many times user tried

int gusCorr=0;//how many correct number have been guessed

int chnsLft; //chance counter(how many chances left)

int digit=0; //digit of the user guesses

char guess=0; //the number user guesses

char pswd[SIZE]={};//the password store in the array

vector<int> inputDg(SIZE,5);//the digits finished

//introduce the game

introduce();

//get a random 4-digit password and put it in array

for(int i=0;i<SIZE;i++) {

pswd[i]=rand()%10+'0';

}

dash=toDash(SIZE);//get the dash

chnsLft=TOTCHNS;

//Use for loop get the password in strings

for(int i=0;i<SIZE;i++) {

answer+=pswd[i];

}

//game begins

while(chnsLft>0&&gusCorr<SIZE) {

//Prompt user for the guess

cout<<endl;

cout<<"The password now looks like this: "<<dash<<endl;//Output dash

cout<<"You have "<<chnsLft<<" chances left"<<endl;//output chances left

ask(guess,digit);//Prompt user for guess

char result=check(guess,digit,pswd,SIZE);//check the guess

switch(result) {

case'1': { //if the number and place both are correct

replace(dash,guess,digit);//replace of the correct digit

if(inside(inputDg,digit)) { //if user have finished that digit

cout<<"You already finish this digit,"

<<"try other digits"<<endl;

} else {//user didn't finish this digit

inputDg.push\_back(digit-1);//record the digit which has been finished to vector

cout<<"Your guess is correct."<<endl;

gusCorr++;

}

break;

}

case'2': { //if number is correct but digit is wrong

cout<<"This is the correct number but in wrong place."<<endl;

chnsLft--;

break;

}

case'3': { //if number and digit both wrong

cout<<"Wrong number and wrong place."<<endl;

chnsLft--;

break;

}

default:;

}

times++;//keep track of how many times user have input

}

if(chnsLft==0) { //No chances left for player

cout<<"You lost"<<endl;

output<<"You lost"<<endl;

}

if(gusCorr==SIZE) { //when 4 digits have been guessed correctly

output<<"You win this game after "<<times<<" tries"<<endl;

cout<<"You win this game after "<<times<<" tries"<<endl;

}

cout<<"The answer is "<<answer<<endl;

output.close();

//Exit stage right

return 0;

}

string toDash(int size) {

string dashed="";

for(int i=0;i<size;i++) {

dashed+="-";

}

return dashed;

}

void introduce() {

cout<<"\*\*\*\*\*\*Welcome to Bomb password\*\*\*\*\*\*"<<endl;

cout<<"In this game, you should guess the password in order to defuse the bomb"<<endl;

cout<<"First, you will input a number that you guess"<<endl;

cout<<"Then, you will input the digit of this number"<<endl;

cout<<"The digit of the number from left to right is 1,2,3,4"<<endl;

cout<<"After you input these two information, the computer will tell you"<<endl;

cout<<"whether the number and digit are correct"<<endl;

cout<<"Attention: some digits of password may be the same number"<<endl;

cout<<"Press Enter to start the game";

cin.ignore();

}

void ask(char& guess,int& digit) {

do {

cout<<"Please input a number you guess"<<endl;

cout<<"If you need sample for input, type \'s\'"<<endl;

cin>>guess;

cin.ignore();

if(guess=='s'||guess=='S') //when player need sample

sample(); //output sample via ifstream

else if(guess<48||guess>57)

cout<<"Invalid input"<<endl<<endl;

} while(guess<48||guess>57);

do {

cout<<"Please input the digit of this number"<<endl;

cin>>digit;

cin.ignore();

if(digit<1||digit>4)

cout<<"Invalid input"<<endl<<endl;

} while(digit<1||digit>4);

}

char check(char guess,int digit,const char pswd[],int size) {

//Because array counts from 0,but digit from left to right is 1,2,3,4,so it need digit-1 for array

if(guess==pswd[digit-1]) { //when guess and digit are correct

return '1';

} else if(indexOf(guess,pswd,size)) { //number is right but digit is wrong

return '2';

} else { //both are wrong

return '3';

}

}

//return whether the char is in the array

bool indexOf(char x,const char pswd[],int size) {

bool temp=false;

for(int i=0;i<size;i++) {

if(pswd[i]==x)

temp=true;

}

return temp;

}

void replace(string& dash,char guess,int digit) {

string part1=dash.substr(0,(digit-1));

string part2=dash.substr(digit);

dash=part1+guess+part2;

}

//return whether is digit has been finished

bool inside(const vector<int> inputDg,int digit) {

bool temp=false;

for(int i=0;i<inputDg.size();i++) {

if(inputDg[i]==(digit-1))

temp=true;

}

return temp;

}

//display sample via file

void sample() {

string str;

ifstream input;

input.open("Sample.dat");

while(input>>str) {

if(str=="Now"||str=="If"||str=="You"||str=="<")

cout<<endl;

cout<<str<<' ';

}

cout<<endl<<endl;

input.close();

}