Description:
This program is a simple command-line crossword game. The way to play it is simple. First there will be a line ask for the crossword list which is a txt file containing the numbers, answers and questions for the crossword. After loading the text file, the game will ask you to choose a number. Then there will be the question corresponding to each number. If you enter the right answer, the puzzle will be generated successfully. Otherwise you need to enter again.
This game uses linux system, which means we could test it on lab computers.

Annotated bibliography:
At first we decided to find some outside source to make our puzzle or command line looks better by adding some color effect for the command lines and frames for the puzzle. However, it was my partner's job to do it. Since he dropped the course, I didn't have any outside source for this program.

Bug:
Since it is a simple command-line game, there is no big bug so far. But if you enter the wrong txt file name, you need to start the program again.

Suggestion:
This program can only use the txt file I provided and it cannot load a different puzzle because the method which generate the puzzle is just for this txt file.

This whole program is written by myself. My partner's job is to find the outside resources to decorate the interface and puzzle. Since he dropped the course early, I didn't have time to finish that. I'm sorry for that.

crossword.cpp:

```
#include <iostream>
#include <unordered_map>
#include <map>
#include <string>
#include <fstream>
#include <sstream>
#include "crossword.hpp"
using namespace std;

//load the crossword list file
CrosswordList::CrosswordList(string crossword) {
    ifstream ifs(crossword+".txt");
    if(!ifs.is_open()) {
        cout << "ERROR: Unable to open crossword list file " + crossword + ".txt" << endl;
        return;
    }
    string line,answer,question;
    int answerNum;
    while(getline(ifs,line)) {
        stringstream stst;
        stst << line;
        stst >> answerNum >> answer;
        getline(stst,question);
        answers[answerNum] = {answer, question};
    }

    //generate a blank puzzle first
    for (int i = 0; i < 21; i++) {
        for (int j = 0; j < 20; j++) {
            puzzle[i][j] = ' ';
        }
    }
}

//print the question of the number which user chooses
void CrosswordList::printCrossWordList(int wordNum) {
    int answerNum = wordNum;
    string question = answers[answerNum].second;
    string answer = answers[answerNum].first;
    //only print out the number and the question
    cout<<answerNum<<question<<endl;
}
```

```
//check if the input answer is correct, if correct, return true, else return false
bool CrosswordList::checkAnswer(int wordNum, string answer, bool correct) {
    int answerNum = wordNum;
    if (answer != answers[answerNum].first) {
        cout<<"Sorry, wrong answer. Please enter again."<<endl;
        correct = false;
    }
    else{
        cout<<"Congratulations! You're right!"<<endl;
        cout<<"===================="<<endl;
        correct = true;
    }
    return correct;
}

//generate the puzzle based on the word users enter
void CrosswordList::creatPuzzle(int wordNum) {
    string answer = answers[wordNum].first;
    switch(wordNum) {
        case 1:
            for(int i = 0; i < 12; i++) {
                puzzle[i][4] = answer[i];
            }
            break;
        case 2:
            for(int i = 0; i < 9; i++) {
                puzzle[1][i] = answer[i];
            }
            break;
        case 3:
            for(int i = 0; i < 8; i++) {
                puzzle[1][12+i] = answer[i];
            }
            break;
        case 4:
            for(int i = 0; i < 8; i++) {
                puzzle[1+i][16] = answer[i];
            }
            break;
        case 5:
            for(int i = 0; i < 7; i++) {
                puzzle[3][i] = answer[i];
            }
            break;
```

```
case 6:
   for(int i = 0; i < 9; i++) {
      puzzle[3][10+i] = answer[i];
   }
   break;
case 7:
   for(int i = 0; i < 7; i++) {
      puzzle[5][i] = answer[i];
   }
   break;
case 8:
   for(int i = 0; i < 12; i++) {
      puzzle[5+i][2] = answer[i];
   }
   break;
case 9:
   for(int i = 0; i < 8; i++) {
      puzzle[6+i][12] = answer[i];
   }
   break;
case 10:
   for(int i = 0; i < 7; i++) {
      puzzle[7][10+i] = answer[i];
   }
   break;
case 11:
   for(int i = 0; i < 8; i++) {
      puzzle[7+i][18] = answer[i];
   }
   break;
case 12:
   for(int i = 0; i < 8; i++) {
      puzzle[10+i][6] = answer[i];
   }
   break;
case 13:
   for(int i = 0; i < 11; i++) {
      puzzle[11][4+i] = answer[i];
   }
   break;
case 14:
   for(int i = 0; i < 11; i++) {
      puzzle[13][9+i] = answer[i];
   }
```

```
            break;
        case 15:
            for(int i = 0; i < 9; i++) {
                puzzle[15][5+i] = answer[i];
            }
            break;
        case 16:
            for(int i = 0; i < 6; i++) {
                puzzle[15+i][11] = answer[i];
            }
            break;
        case 17:
            for(int i = 0; i < 10; i++) {
                puzzle[17][8+i] = answer[i];
            }
            break;
        case 18:
            for(int i = 0; i < 7; i++) {
                puzzle[20][6+i] = answer[i];
            }
    }
}

//print out the puzzle
void CrosswordList::printPuzzle() {
    cout<<"--------------------"<<endl;
    for (int i = 0; i < 21; i++) {
        for (int j = 0; j < 20; j++) {
            cout<<puzzle[i][j];
        }
        cout<<endl;
    }
    cout<<"--------------------"<<endl;

}

//get the size of the list in order to control the times for user to enter the answer
int CrosswordList::getSize() {
    this->size = answers.size();
    return this->size;
}

int main(){
    string textName;
```

```
    int wordNum;
    string word;
    bool correct = false;
    cout<<"Welcome to crossword!"<<endl;
    cout<<"Please enter your crossword list file:";
    cin>>textName;
    CrosswordList x(textName);

    int count = x.getSize();
    //make sure user could enter all the correct answer before the program ends
    for (int i = 0; i < count;) {
        cout<<"====================="<<endl;
        cout<<"Please choose your number: ";
        cin>>wordNum;
        if(wordNum > count) {
            cout<<"Sorry the biggest number is "<<" "<<count<<" "<<"Please choose your number
again: ";
            cin>>wordNum;
        }
        x.printCrossWordList(wordNum);
        cout<<"Please enter your answer (lower case): ";
        cin>>word;
        correct = x.checkAnswer(wordNum, word, correct);
        //generate a new puzzle only if the answer is correct
        if (correct == true) {
            x.creatPuzzle(wordNum);
            x.printPuzzle();
            i++;
        }
    }
    cout<<"Congratulations! You complete this puzzle!"<<endl;
    cout<<"========================================="<<endl;
}
```

crossword.hpp:

```
#include <iostream>
#include <unordered_map>
#include <map>
#include <string>
#include <fstream>
using namespace std;

class CrosswordList {
   protected:
      map<int,pair<string,string>> answers; //list for the answers and questions
      int size; //list size
      string puzzle[21][20]; //String array for generating a puzzle
   public:
      CrosswordList(string crossword); //load the list of numbers, answers and questions
      void printCrossWordList(int wordNum); //print out the number and question
      bool checkAnswer(int wordNum, string answer, bool correct); //check if the input answer is
correct
      void creatPuzzle(int wordNum); //generate the puzzle
      void printPuzzle(); //print the puzzle
      int getSize(); //get the size of the list
};
```