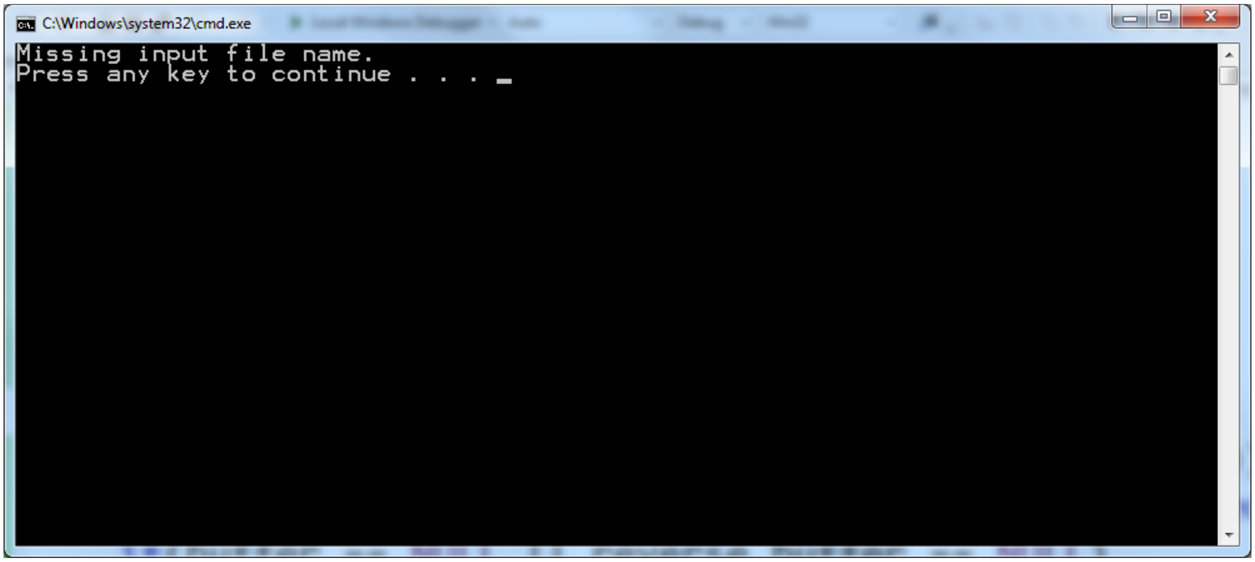
Create a project called Daily25. Add a C source file to the project named daily25.c.

Write a program that opens a file reads all of the strings in the file separated by whitespace, prints the strings on the screen followed by the two characters -> and then prints the string in reverse followed by a new line. The file name will be given as the first command line argument after the name of the executable (argv[1]). You will malloc enough space to hold 50 characters, one of which is the NULL terminator and use a function called read_string as we created in class to read in each string. Your function should skip over all leading whitespace characters and stop reading when you hit a whitespace character or EOF or fill the buffer you are working on.

Your program should print an appropriate message and exit if the user forgets to give a command line argument for the file name or if the file does not exist in the working directory.

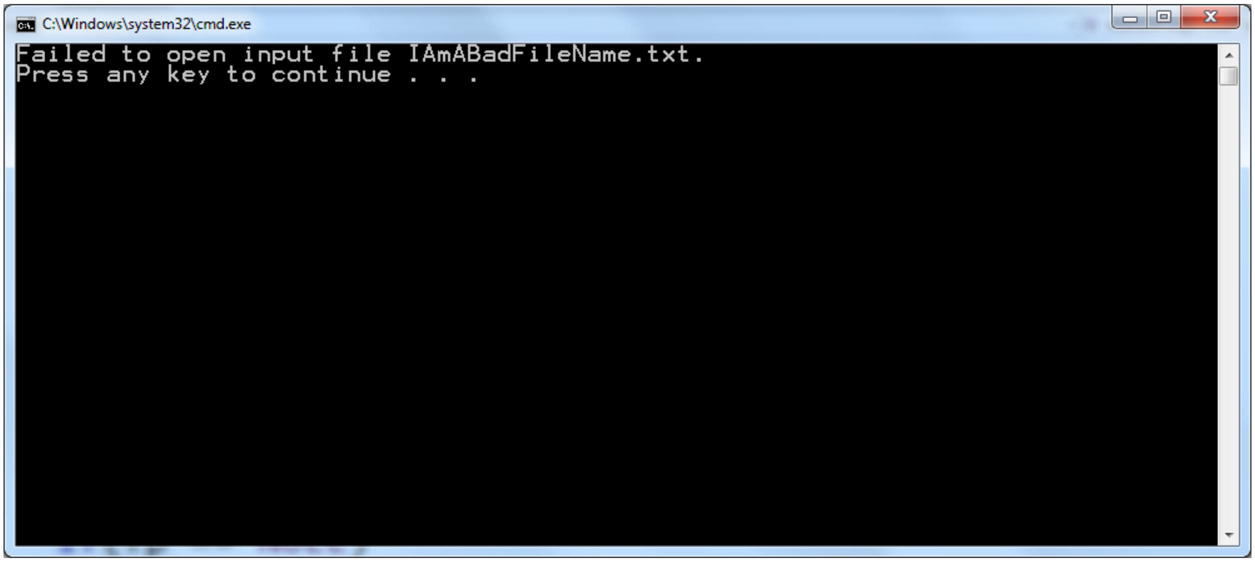
Your output should look something like the following:

Running with no command line arguments:



```
Missing input file name.
Press any key to continue . . .
```

Run with an invalid name:



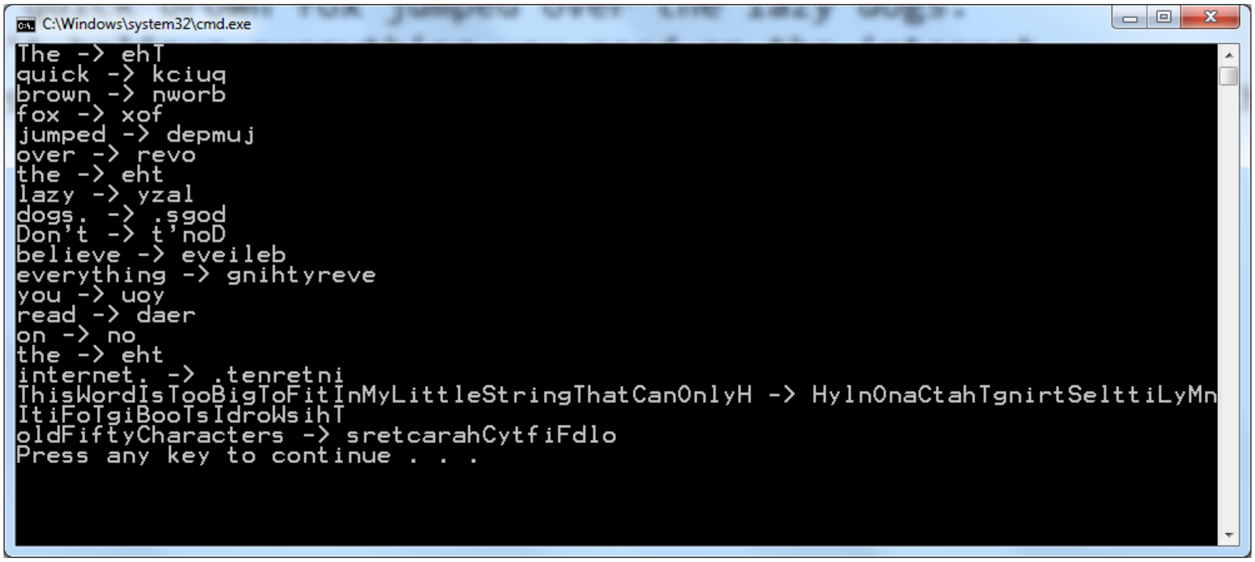
```
Failed to open input file IAmABadFileName.txt.
Press any key to continue . . .
```

And finally run with a file that contains the following text:

The quick brown fox jumped over the lazy dogs.

Don't believe everything you read on the internet.

ThisWordIsTooBigToFitInMyLittleStringThatCanOnlyHoldFiftyCharacters



```
The -> ehT
quick -> kciuq
brown -> nworb
fox -> xof
jumped -> depmuj
over -> revo
the -> eht
lazy -> yzal
dogs. -> .sgod
Don't -> t'noD
believe -> eveileb
everything -> gnihtyreve
you -> uoy
read -> daer
on -> no
the -> eht
internet. -> .tenretni
ThisWordIsTooBigToFitInMyLittleStringThatCanOnlyH -> HylnOnaCtahTgnirtSelttiLyMn
ItiFoTgiBooTsIdroWsihT
oldFiftyCharacters -> sretcarahCytfiFdlo
Press any key to continue . . .
```

At the top of your program you should have a comment section that follows the below format:

```
/***********************************************
     Author: <insert your name>
     Date: 11/14/2014

     Purpose: <Insert a short description of what
          your program does here.>
     Time Spent: <Insert how much time you spent
          on the assignment here>
***********************************************/
```