The words above represent a simple substitution cipher
- Each letter mapped to one other letter, no inconsistencies
- Often used in cryptogram puzzles (newspaper, online, ...)
- How can we write a computer program to solve this?

Ideas for solving the problem? Benchmark/ballpark idea to accept (or not)

Problems on the horizon?

One possible solution in docrypto.cpp
- Study this for an example of backtracking
  - Similar to N queens: make move, recurse, undo as needed
  - What’s a move in this problem?

- Illustrates a few C++ and OO concepts
  - Static variables and functions: belong to class not object
  - Also called “class variables”, don’t need object to access
  - Must be careful when initializing static variables because order of initialization can be important

- See WordSource object shared by all CryptoMap objects, how and when is the WordSource initialized?

Heuristics
- A heuristic is a rule of thumb, doesn’t always work, isn’t guaranteed to work, but useful in many/most cases
  - Search problems that are “big” often can be approximated or solved with the right heuristics

- What heuristic is good for cryptograms?
  - Solve small words first
  - Solve large words first
  - Do something else?

- What other optimizations/improvements can we make?
  - See program, cryptomap.cpp and docrypto.cpp

Solving problems and coding
- Determine if proposed solution is viable
  - Does it work algorithmically in reasonable time?
  - Can you code it in reasonable time?

- How to develop the program
  - Develop a core, test it, make it work
  - Grow the core and test each piece as it’s grown
  - Test classes in isolation from each other, not as part of the whole program (why?)

- Do code/design walkthroughs with friends, teachers, mentors
  - Egoless programming: you’re not a bozo