Numbers to Words

This project will deal with conditionals, loops, and random numbers.

Turn in

- On Titanium:
  - Your .cpp file
  - Screenshot(s) of 5 runs of your program
- In class: These same two documents, printed and stapled

Rules

- If your solution uses arrays then it must not use loops (i.e., it must be fully recursive).
- Functions are optional, but recommended.
- Please try to design the algorithm on your own, or with other students, or with help from us.
- You may, randomly, be required to explain and justify your algorithm, and any code you write. If you're worried you might forget something, please comment your code so that you don't.

Suggestions

- Look up the `rand()` and `srand()` functions.
- Recall the modulus operator (`%`), integer division, and switch statements.
- Think carefully about which type of loop would most cleanly do what you want. If you're not sure, and you have time, try all three.
- Don't be afraid to use the member functions of `std::string`, if you need them: `http://en.cppreference.com/w/cpp/string/basic_string`.
- Be sure to include the header files declaring the functions that you're trying to use!

Requirements

In comments at the top of your code, please include:

- Your name.
- The assignment (i.e., "Project 4").
- Your development environment (including version number).
- A list of any nontrivial references you consulted while writing your solution.
- A list of any notes you have regarding the assignment or your solution.
  - This should include a short discussion of any issues you ran into while using `rand()`, or testing your program. Are there any limitations of `rand()` that you happened to notice?
- A pseudocode description of your algorithm, written as if to explain your ideas to an average classmate without previous knowledge of the assignment.

Your code must:

- Use standard C++ functions to obtain two pseudo-random integers that are *not* the same on every run of the program.
  - The first should be called `start`, and should be between -2^{31} and $2^{31} - 1$, inclusive.
  - The second should be called `offset`, and should be between 2 and 8, inclusive.
- Print out a message indicating the value of `start`.
- Print out a message indicating the value of `offset`.
- Print out (each on its own line) the English form of the integer `start`, and every third integer after that until `offset` integers have been printed.

Sample Output

```
Last login: Tue Mar 18 00:32:13 on ttys000
ben@ben-mba:~/Desktop/project-4$ clang++ solution--project-4--num2word.cpp -o solution
ben@ben-mba:~/Desktop/project-4$ ./solution
start: -10794173
offset: 6
negative ten million seven hundred ninety four thousand one hundred seventy three
negative ten million seven hundred ninety four thousand one hundred seventy
negative ten million seven hundred ninety four thousand one hundred sixty seven
negative ten million seven hundred ninety four thousand one hundred sixty four
negative ten million seven hundred ninety four thousand one hundred sixty one
negative ten million seven hundred ninety four thousand one hundred fifty eight
ben@ben-mba:~/Desktop/project-4$ ./solution
start: 858219920
offset: 8
eight hundred fifty eight million two hundred nineteen thousand nine hundred twenty
eight hundred fifty eight million two hundred nineteen thousand nine hundred twenty three
eight hundred fifty eight million two hundred nineteen thousand nine hundred twenty six
eight hundred fifty eight million two hundred nineteen thousand nine hundred twenty nine
eight hundred fifty eight million two hundred nineteen thousand nine hundred thirty two
eight hundred fifty eight million two hundred nineteen thousand nine hundred thirty five
eight hundred fifty eight million two hundred nineteen thousand nine hundred thirty eight
eight hundred fifty eight million two hundred nineteen thousand nine hundred forty one
ben@ben-mba:~/Desktop/project-4$ ./solution
start: 1423170418
offset: 6
one billion four hundred twenty three million one hundred seventy thousand four hundred eighteen
one billion four hundred twenty three million one hundred seventy thousand four hundred twenty one
one billion four hundred twenty three million one hundred seventy thousand four hundred twenty four
one billion four hundred twenty three million one hundred seventy thousand four hundred twenty seven
one billion four hundred twenty three million one hundred seventy thousand four hundred thirty
one billion four hundred twenty three million one hundred seventy thousand four hundred thirty three
ben@ben-mba:~/Desktop/project-4$ ./solution
start: -123112518
offset: 8
negative one hundred twenty three million one hundred twelve thousand five hundred eighteen
negative one hundred twenty three million one hundred twelve thousand five hundred fifteen
negative one hundred twenty three million one hundred twelve thousand five hundred twelve
negative one hundred twenty three million one hundred twelve thousand five hundred nine
negative one hundred twenty three million one hundred twelve thousand five hundred six
negative one hundred twenty three million one hundred twelve thousand five hundred three
negative one hundred twenty three million one hundred twelve thousand five hundred
negative one hundred twenty three million one hundred twelve thousand four hundred ninety seven
ben@ben-mba:~/Desktop/project-4$ ./solution
start: 970538265
offset: 4
nine hundred seventy million five hundred thirty eight thousand two hundred sixty five
nine hundred seventy million five hundred thirty eight thousand two hundred sixty eight
nine hundred seventy million five hundred thirty eight thousand two hundred seventy one
nine hundred seventy million five hundred thirty eight thousand two hundred seventy four
ben@ben-mba:~/Desktop/project-4$
```