Write a C++ program that converts dollar amounts from numeric form into its English equivalent, similar to what would be seen on a check. Your program should produce output *exactly* like the example below.
_____

```
Amount? 143

One Hundred Forty Three Dollars and No Cents

Amount? 2.345

Two Dollars and 35 Cents

Amount? 0.1

No Dollars and 10 Cents

Amount? 1.01

One Dollar and 1 Cent

Amount? 123456789.12

One Hundred Twenty Three Million Four Hundred Fifty Six Thousand Seven Hundred
Eighty Nine Dollars and 12 Cents

Amount? 0
```
_____

Rules:
1. Input numbers will always be between 0 and 1 billion. (use a double to represent this amount)
    a.   Honors section must also play the proper audio files.
2. Stop asking for amounts when 0 is entered.
3. Capitalize as shown in the above example.
4. Put the dollars in words but the cents in numbers (except for No Cents).
5. Round all cents to the nearest penny.
6. Make sure the singulars and plurals are correct (Cent vs. cents, dollar vs. dollars).
7. **The main purpose of this program is to get experience with functions. You will be penalized greatly for redundant code and unclear code, so use functions when appropriate.**
8. Proper indentation and code format.
9. Good variable names.

Don't worry about text being too long for the screen- it's ok for a word to be split in half at the screen edge.

Checks.cpp should be submitted on easel.

Start with this code:

```cpp
#include<iostream>
using namespace std;

const int MAX_LENGTH = 1000;

void displayMoneyAsText( double money );

void main()
{
```

```
        double money;

        do
        {
                cout << "Amount? ";
                cin >> money;

                if( money > 0 )
                {
                        displayMoneyAsText( money);
                }
        }
        while( money > 0 );

}

void displayMoneyAsText( double money )
{
}
```

Rubric:

Rule 1 – works for all numbers in that range	46%
Rule 2	4%
Rule 3	4%
Rule 4	10%
Rule 5	4%
Rule 6	4%
Rule 7	20%
Rule 8	4%
Rule 9	4%

Honors Addition:
An audible version conforming to the same rules listed above should be played.
Sound files can be copied from the Harding network at
        \\cs1\Classes\Comp170\Number Voices for Checks Program
Or downloaded from:
        https://www.harding.edu/dsteil/170/voices.zip

```
#include <windows.h>
#include <iostream>

using namespace std;

//for this to work you must add winmm.lib to
//menu -> Project | Properties
//tree -> Configuration Properties | Linker | Input | Additional Dependencies

void main()
{
        PlaySound(TEXT("Number Voices for Checks Program\\Lumberjack\\one.wav"),
                        NULL, SND_FILENAME | SND_SYNC );
}
```