```
//Answer to question #39042, Programming, C++

#include <iostream>
#include <string>
using namespace std;

int main()
{
	string numStr;// for user entered number
	bool isNegative = false;

	// names for use in output
	string onesName[] = { "one", "two", "three", "four", "five", "six", "seven",
"eight", "nine" };
	string teensName[] = { "ten", "eleven", "twelve", "thirteen", "fourteen",
"fifteen", "sixteen", "seventeen", "eighteen", "nineteen" };
	string tensName[] = { "twenty", "thirty", "forty", "fifty", "sixty", "seventy",
"eighty", "ninety" };
	string illion_preName[] = { "m", "b", "tr", "quadr", "quint", "sext", "sept",
"oct", "non", "dec" };
	string decillion_preName[] = { "un", "duo", "tre", "quattuor", "quin", "sex",
"septen", "octo", "novem" };

	char repeat = 'n';

	do// as long as user wishes to enter number for naming
	{
		cout << "Number = "; cin >> numStr;

		// check for '-' as 1st character
		if (numStr[0] == '-')
		{
			isNegative = true;
			numStr.erase(0, 1);
		}
		else
			isNegative = false;

		// validate entry: check that all characters are digits
		bool isValid = true;
		for (unsigned int i = 0; i < numStr.size(); ++i)
		if (numStr[i] < '0' || numStr[i] > '9')
		{
			isValid = false;
			break;
		}
		if (!isValid)
		{
			cout << "Your entry contains invalid characters." << endl;
			goto repeat;// evil but effective
		}

		// check that number of digits is not too high.
		cout << "power = " << numStr.size() - 1 << endl;
		if (numStr.size() > 66)
		{
			cout << "The number's too damn big! Try again." << endl;
			goto repeat;
		}// validation complete

		// process the validated entry
		while (numStr.size() % 3 != 0)
			numStr = '0' + numStr;// pad the string with leading '0' until size
= multiple of 3

		// print if number is negative
		if (isNegative) cout << "negative ";

		// for each group of 3 digits from most to least significant
		for (unsigned int i = 0; i < numStr.size(); i += 3)
		{
			// skip if all 3 digits == '0'
			if (numStr[i] == '0' && numStr[i + 1] == '0' && numStr[i + 2] ==
'0')
				continue;

			if (numStr[i + 0] > '0')// treat the hundreds place
				cout << onesName[numStr[i + 0] - '0' - 1] << " hundred ";

			if (numStr[i + 1] == '0' || numStr[i + 1] > '1')// treat tens and
ones digits for non-teens case
			{
				if (numStr[i + 1] > '1') cout << tensName[numStr[i + 1] - '0'
- 2] << " ";
				if (numStr[i + 2] > '0') cout << onesName[numStr[i + 2] - '0'
- 1] << " ";
			}
			else// special teens case
				cout << teensName[numStr[i + 2] - '0'] << " ";

			// naming each factor of 1,000
			unsigned int j = (numStr.size() - i) / 3;
			if (j == 2) cout << "thousand ";
			else if (j > 2)
			{

				if (j <= 12) cout << illion_preName[j - 3];// 'xx' before
"illion" cases
				else if (j <= 21) cout << decillion_preName[j - 13] <<
"dec";// 'xx' before "dec" + "illion" cases
				else if (j == 22) cout << "vigint";// special 'xx' before
"vigint" + "illion" case

				cout << "illion ";// the "illion" suffix
			}
		}

	repeat:
		cout << endl << "Repeat? (y/n): ";
		cin >> repeat;
	} while (repeat == 'y');


	cout << endl;
	return 0;
}
```