```
#include <iostream>
#include <fstream>
#include <string>

using namespace std;

string latin_to_roman(int lat_num);

void append_n_times(string& result,
                    string const& a,
                    int n)
{
  while ( n --> 0 )
    result += a;
}
```

```
int main(int argc, char* argv[])
{
  int latin_number;
  ifstream* infile = 0;
  istream* in = &cin;

  if (argc == 2)
  {
    infile = new ifstream(argv[1]);

    if ( ! *infile )
    {
      cerr << "File '" << argv[1]
           << "' could not be opened."
           << endl;
    }
    else
    {
      in = infile;
    }
  }

  while (*in >> latin_number)
  {
    cout << latin_number << " = "
         << latin_to_roman(latin_number)
         << endl;
  }

  delete infile;

  return 0;
}
```

```
struct my_pair
{
  int first;
  string second;
};

string latin_to_roman(int lat_num)
{
  struct my_pair translation[] = {
    {1, "I"},
    {5, "V"},
    {9, "IX"},
    {10, "X"},
    {50, "L"},
    {90, "XL"},
    {100, "C"},
    {500, "D"},
    {900, "CM"},
    {1000, "M"}
  };

  string rom_str;

  const int SIZE = (sizeof(translation) /
                    sizeof(struct my_pair));
  for (int i = SIZE - 1; i >= 0; --i)
  {
    string glyph = translation[i].second;
    int value = translation[i].first;

    if (lat_num >= value)
    {
      append_n_times(rom_str, glyph,
                     (lat_num / value));
      lat_num %= value;
    }
  }
  return rom_str;
}
```