西北大学

面向对象程序设计

上机实验报告

学号：2020115025

班级：计科2班

姓名：薄劲阳

日期：2022年4月19日

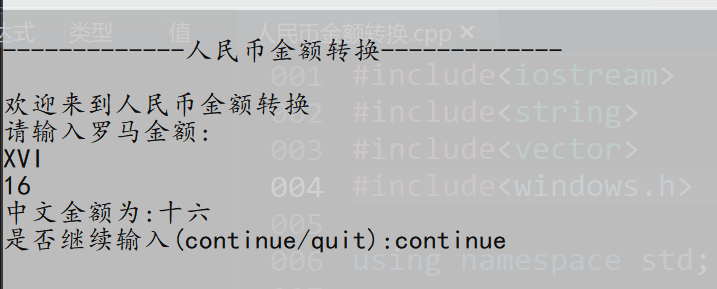
1. 问题分析

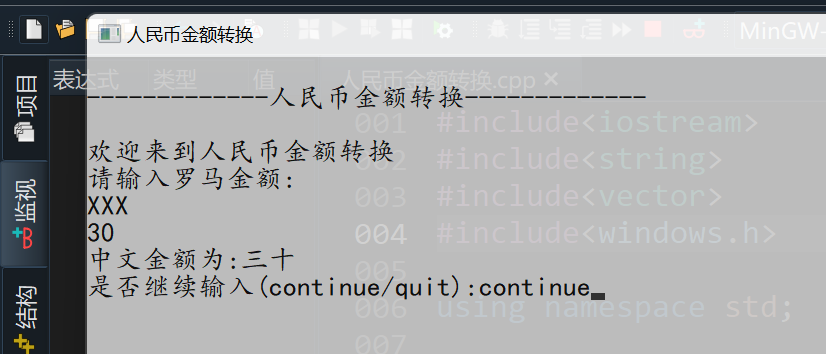
* 程序需要将罗马数字金额转换成中文金额
* 金额转换程序要不断执行，直到用户输入quit结束程序
* 用户输入的罗马数字金额采用C风格字符串存储
* 输出的中文金额采用字符串存储

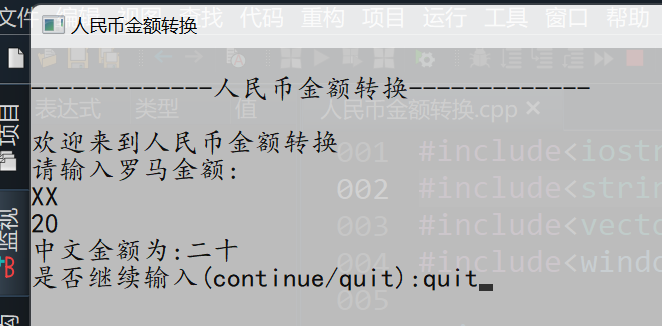
1. 问题建模和流程设计

* 罗马数字的基本规则是：
  + **相同的数字连写、所表示的数等于这些数字相加得到的数、**如：Ⅲ=3；
  + **小的数字在大的数字的右边**、所表示的数等于这**些数字相加得到的数**。 如：Ⅷ=8、Ⅻ=12；
  + **小的数字（限于 Ⅰ、X 和 C）在大的数字的左边**，所表示的数等于**大数减小数得到的数。**如：Ⅳ=4、Ⅸ=9；
  + 正常使用时、连写的数字重复不得超过三次；
  + 在一个数的上面画一条横线、表示这个数扩大 1000 倍。
  + 按照这个规则就是把罗马数字的每个字符翻译成整型数字，然后按照顺序进行判断，从左到右，当后边一个大于前边一个时，就做减法运算，否则就做加法运算。
* 罗马数字转阿拉伯数字
* 阿拉伯数字转中文

三、程序实现和测试







四、实验源码

1. **#include<iostream>**
2. **#include<string>**
3. **#include<vector>**
4. **#include<windows.h>**
6. **using namespace std;**
8. **string AA[] = { "零", "一", "二", "三", "四", "五", "六", "七", "八", "九", "十" };**
9. **string BB[] = { "", "十", "百", "千", "万", "十万", "百万", "千万", "亿" };**
11. **//界面**
12. **void Menu() {**
13. **system("color 70");**
14. **SetConsoleTitle("人民币金额转换");**
15. **cout << "\n-------------人民币金额转换-------------\n" << endl;**
16. **cout << "欢迎来到人民币金额转换" << endl;**
17. **return;**
18. **}**
20. **//罗马转数字**
21. **int RomanToInt(string s) {**
22. **int n = s.length();//罗马数字输入——罗马数字的长度代表有几个字符，字符可以根据其位置进行加减运算**
23. **int res = 0;//阿拉伯数字**
24. **int\* num = new int[n];//动态分配数组**
25. **for (int i = 0; i < n; i++) {**
26. **switch (s[i]) {**
27. **case 'M':**
28. **num[i] = 1000;//M-1000**
29. **break;**
30. **case 'D':**
31. **num[i] = 500;//D-500**
32. **break;**
33. **case 'C':**
34. **num[i] = 100;//C-100**
35. **break;**
36. **case 'L':**
37. **num[i] = 50;//L-50**
38. **break;**
39. **case 'X':**
40. **num[i] = 10;//X-10**
41. **break;**
42. **case 'V':**
43. **num[i] = 5;//V-5**
44. **break;**
45. **case 'I':**
46. **num[i] = 1;//I-1**
47. **break;**
48. **default:**
49. **break;**
50. **}**
51. **}**
52. **//计算罗马数字**
53. **for (int i = 0; i < n - 1; i++) {**
54. **if (num[i] < num[i + 1])**
55. **//如果后者比前者大，意味着要用后项减前项，也可以转换成数字和加该前项的负数**
56. **//IV 可以转换成 -I+V**
57. **res -= num[i];**
58. **else//前项大于等于后项，直接加**
59. **res += num[i];**
60. **}**
61. **res += num[n - 1];//**
62. **return res;**
63. **}**
64. **//数字转中文**
65. **string IntToChinese(int num) {**
66. **string res = "";//存储中文名**
67. **string numstr = to\_string(num); //直接把00123这种前面的0去掉了**
68. **int k = numstr.length();//存储长度**
69. **for (int i = 0; i < numstr.length(); i++) {**
70. **int tmp = numstr[i] - '0';//tmp存储当前位置的数字**
71. **if (0 == tmp) {//当前位置为0，上一位置也是零，或者最后到数字最后一个位置**
72. **if ('0' == numstr[i - 1] || i == numstr.length() - 1)**
73. **continue;**
74. **else**
75. **res = res + AA[tmp];//进行连接操作——转换为中文**
76. **} else {**
77. **res = res + AA[tmp];//进行连接操作**
78. **if (numstr.length() == 2 && numstr[0] == '1' && i == 0) {**
79. **res.erase(0); //12读作十二，把1删掉**
80. **}**
81. **res = res + BB[k - i - 1];**
82. **}**
83. **}**
84. **return res;**
85. **}**

88. **int main() {**
90. **string flag;**
91. **do {**
92. **Menu();//菜单打印**
93. **string RomanMoney;//存储罗马数字**
94. **string ChineseMoney;//存储中文数字**
95. **//罗马数字输入**
96. **cout << "请输入罗马金额:" << endl;**
97. **cin.sync();//清除缓冲区**
98. **getline(cin, RomanMoney);**
99. **//罗马数字转数字**
100. **int DigitMoney = RomanToInt(RomanMoney);//罗马转数字**
101. **cout << DigitMoney << endl;**
102. **ChineseMoney = IntToChinese(DigitMoney);//数字转中文**
103. **cout << "中文金额为:" << ChineseMoney << endl;**
104. **cout << "是否继续输入(continue/quit):";**
105. **cin >> flag;**
106. **system("cls");//翻页**
108. **} while (flag != "quit");**
110. **return 0;**
112. **}**