

## C++プログラミング応用演習 2

### 第二回レポート課題

ht17a053 田仲凌真

[ht17a053@oecu.jp](mailto:ht17a053@oecu.jp)

## 1.作成したプログラムのソースコード

```
1.// ex-2-main.cpp
2.#include <iostream>
3.#include <string>
4.using namespace std;
5.
6.int toint(string s) {
7.    char c[4];
8.    int v[4];
9.    int sum = 0;
10.   int d = 1;
11.   for (int i = 0; i < s.size(); i++) {
12.       if ('0' <= s[i] && s[i] <= '9') {
13.           d = s[i] - '0';
14.       }
15.       else {
16.           for (int j = 0; j < 4; j++) {
17.               if (s[i] != c[j]) { continue; }
18.               sum += v[j] * d;
19.               d = 1;
20.               break;
21.           }
22.       }
23.   }
24.   return sum;
25.}
26.int main() {
27.    char c[4] = { 'm', 'c', 'x', 'i' };
28.    int v[4] = { 1000, 100, 10, 1 };
29.    int N;
30.    int s1, s2;
31.    cin >> N;
32.    while (N--) {
33.        cin >> s1 >> s2;
34.        int sum = int(s1) + int(s2);
35.        for (int i = 0; i < 4; i++) {
```

```

36.                int d = 0;
37.                while (sum >= v[i]) { sum -= v[i]; d++; }
38.                if (d > 1) { cout << d; }
39.                if (d > 0) { cout << c[i]; }
40.            }
41.            cout << "結果" << s1+s2 << endl;
42.        }
43.
44.    return 0;
45.}

```

ソースコード“ex-2-main.cpp”

## 2. ソースコードの解説

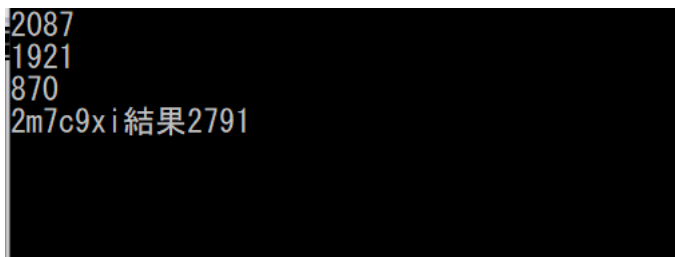
7 行目、8 行目で文字列と数値を宣言する。9 行目で合計数の初期値を宣言する。10 行目で for 文を抜け出すために使う数値の初期値を宣言する。

11 行目から 23 行目は for 文のループを使用している。11 行目から 14 行目は  $i$  が 0、 $i$  が  $s.size$  より小さい間、もし  $s[i]$  が 0 以上かつ、9 以下ならば  $d$  に  $s[i]-0$  を代入する。15 行目から 22 行目は、そうでない場合、 $j$  が 4 未満の場合に、非等価である場合、if ぶんのなかの計算を実行することをさす。

26 行目からは main 文。27 行目は文字列の宣言。28 行目は数列の宣言を行う。29 行目は入力するための数字を代入するために宣言。30 行目も同様である。31 行目はコンソール上で  $N$  に直接代入できるように書き加えられる。32 行目から 40 行目で  $N$ ,  $S_1$ ,  $S_2$  の数値を足し、mcxi 文字列に変換することのできる main 文の挙動を記述する。41 行目はコンソール上に表示される結果の出力である。

これで入力した数値を MCXI 文字列に変換するソースコードを記述することができる。

## 3. 実行結果



```

2087
1921
870
2m7c9xi 結果2791

```

## 4. 考察

当初に課題として課されていた要素についてはプログラミングにより実装することができた。しかし、授業中において、さらに発展させた場合のプログラムテストを兼ねたプログラミングにおいては授業外で一人の力で記述することができなかった。テストコードを実装するために、様々な方法が考えられるが、テストコードを利用したブラックボックス

テスト、あるいはホワイトボックステストが必要なものと思われる。

## 5.感想

考察で記述した通り、個人の力でテストコードを記述することができなかったため、こちらの方はプログラムをかける友人の力を借りながら理解した。今回の課題においては課題2の条件で提出しているのでテストコードを付与したプログラムではないが、ある程度の理解を得ることのできたと思う。就職先でC++を使うかどうか、別のプログラミング言語を使うか確定的ではないものの、今回の授業。また、就職までの最後の一年間で自主的にスキルを身に着けることでいろいろなものを補っていきたいと思う。