HW8 Part 3 coding

108072147 林汶螢

1.

```cpp
#include <iostream>
#include <string.h>
using namespace std;
class Number{
    private:
        int number;
        string lessThan20[20] = {
            "zero", "one", "two", "three", "four", "five", "six", "seven",
            "eight", "nine", "ten", "eleven", "twelve", "thirteen", "fourteen",
            "fifteen", "sixteen", "seventeen", "eighteen", "nineteen" };
        string ty[10] = {"", "",
            "twenty", "thirty", "forty", "fifty",
            "sixty", "seventy", "eighty", "ninety" };
        string hundred = "hundred";
        string thousand = "thousand";
    public:
    Number(int num){
        this->number = num;
    }
    string translate(int num){
        string rtn;
        int S = num%10, T = (num%100 - S)/10;
        int Th = (num%10000 - num%1000)/1000, H = (num%1000 - T)/100;
        if( Th > 0){
            rtn += this->lessThan20[Th];
            rtn += " ";
            rtn += this->thousand;
            rtn += " ";
        }
        if( H > 0){
            rtn += this->lessThan20[H];
            rtn += " ";
            rtn += this->hundred;
            rtn += " ";
        }
        if( T > 0){
            if( T >= 2){
                rtn += this->ty[T];
                rtn += " ";
            }
            else {
                rtn += this->lessThan20[num%100];
                rtn += " ";
                return rtn;
            }
        }
        if(S > 0){
            rtn += lessThan20[S];
            rtn += " ";
        }
        return rtn;
    }
};

int main(){
    int num = -1;
    while(num<0||num>9999){
        cout<<"Please enter a number. (0~9999)  ";
        cin>>num;
        cout<<endl;
    }
    Number number(num);
    cout<<number.translate(num)<<endl;
    return 0;
}
```

Output

```
Please enter a number. (0~9999)  1853

one thousand eight hundred fifty three
```

2.

2.cpp > main(int, char * [])

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

class DayOfYear{
    private:
    int day;
    const string month[12]={"January", "February", "March",
                            "April", "May", "June", "July",
                            "August","September","October",
                            "November","December" };
    public:
    DayOfYear (int num){
        this -> day = num;
    }
    void print(int num){
        cout<<"Day "<<this->day<<" would be "<<this->transition(num)<<endl;
    }
    string transition(int num){
        string rtn;
        if (num<=31){
            stringstream ss;
            ss<<num;
            string date;
            ss>>date;
            ss.clear();
            rtn += month[0] + " " + date;
            return rtn;
        }
        if (num>31 && num<=59){
            stringstream ss;
            int cal = (num-31);
            ss<<cal;
            string date;
            ss>>date;
            ss.clear();
            rtn += month[1] + " " + date;
            return rtn;
        }
        if (num>59 && num<=90){
            stringstream ss;
            int cal = (num-59);
            ss<<cal;
            string date;
            ss>>date;
            ss.clear();
            rtn += month[2] + " " + date;
            return rtn;
        }
        if (num>90 && num<=120){
            stringstream ss;
            int cal = (num-90);
            ss<<cal;
            string date;
```

```
            ss>>date;
            ss.clear();
            rtn += month[3] + " " + date;
            return rtn;
        }
        if (num>120 && num<=151){
            stringstream ss;
            int cal = (num-120);
            ss<<cal;
            string date;
            ss>>date;
            ss.clear();
            rtn += month[4] + " " + date;
            return rtn;
        }
        if (num>151 && num<=181){
            stringstream ss;
            int cal = (num-151);
            ss<<cal;
            string date;
            ss>>date;
            ss.clear();
            rtn += month[5] + " " + date;
            return rtn;
        }
        if (num>181 && num<=212){
            stringstream ss;
            int cal = (num-181);
            ss<<cal;
            string date;
            ss>>date;
            ss.clear();
            rtn += month[6] + " " + date;
            return rtn;
        }
        if (num>212 && num<=243){
            stringstream ss;
            int cal = (num-212);
            ss<<cal;
            string date;
            ss>>date;
            ss.clear();
            rtn += month[7] + " " + date;
            return rtn;
        }
        if (num>243 && num<=273){
            stringstream ss;
            int cal = (num-243);
            ss<<cal;
            string date;
            ss>>date;
            ss.clear();
            rtn += month[8] + " " + date;
            return rtn;
```

```
109             return rtn;
110         }
111         if (num>273 && num<=304){
112             stringstream ss;
113             int cal = (num-273);
114             ss<<cal;
115             string date;
116             ss>>date;
117             ss.clear();
118             rtn += month[9] + " " + date;
119             return rtn;
120         }
121         if (num>304 && num<=334){
122             stringstream ss;
123             int cal = (num-304);
124             ss<<cal;
125             string date;
126             ss>>date;
127             ss.clear();
128             rtn += month[10] + " " + date;
129             return rtn;
130         }
131         if (num>334 && num<=365){
132             stringstream ss;
133             int cal = (num-334);
134             ss<<cal;
135             string date;
136             ss>>date;
137             ss.clear();
138             rtn += month[11] + " " + date;
139             return rtn;
140         }
141     }
142 };
143 int main(int argc, char* argv[]){
144     int times,num;
145     cout<<"Please enter how many times u want to transit. ";
146     cin>>times;
147     while(times--){
148         cout<<"Enter the day u want to know: ";
149         cin>>num;
150         DayOfYear dayofyear(num);
151         dayofyear.print(num);
152     }
153     return 0;
154 }
```

Output

```
Please enter how many times u want to transit. 3
Enter the day u want to know: 15
Day 15 would be January 15
Enter the day u want to know: 37
Day 37 would be February 6
Enter the day u want to know: 62
Day 62 would be March 3
```

3.

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

class Pstring : public string{
    private:

    public:
    Pstring() : string(){}
    Pstring(const char* s) : string(s){}
    Pstring(const string& str) : string(str){}
    bool isPalindrome(){
        int size = this->size()/2;
        for(int i = 0; i < size; i++) {
            if(this->operator[](i) != this->operator[](this->size() - 1 - i)) return false;
        }
        return true;
    }
};

int main(int argc, char* argv[]){
    Pstring pstring;
    string s;
    while(true) {
        cout << "Try to input something." << endl;
        getline(cin, pstring);
        if(pstring.isPalindrome()){
            cout << "It is Palindrome!!" << endl << endl;
        } else {
            cout << "It is not Palindrome!!" << endl << endl;
        }
    }
    return 0;
}
```

Output

```
aloola
It is Palindrome!!

Try to input something.
rteij
It is not Palindrome!!

Try to input something.
oial[
It is not Palindrome!!
```

4.

```cpp
#include <iostream>
#include <iomanip>
using namespace std;
class DivSales{
    private:
        int season[4];
        int sum;
    public:
    DivSales(){
        int i=4;
        while(i--) this->season[i] = 0;
        this-> sum = 0;
    }
    void sales(int a, int b, int c, int d){
        season[0] = a;
        season[1] = b;
        season[2] = c;
        season[3] = d;
    }
    int getsales(int num){
        return season[num];
    }
    int getsum(){
         for(int i = 0; i < 4 ; i++){
                this->sum += this->season[i];
            }
        return this->sum;
    }
};
int main(int argc, char* argv[]){
    DivSales divsales[6];
    int input[4];
    for(int i = 0; i < 6; i++){
        cout<<"==========division sale==========\n";
        cout<<"-              # "<<i+1<<"              -\n";
        for (int j = 0; j < 4; j++){
            cout<<"Sales for season "<<j+1<<": ";
            cin>>input[j];
        }
        cout<<"================================\n";
        divsales[i].sales(input[0], input[1], input[2], input[3]);
    }
    cout<<"\n=============Final report=============\n";
    for(int i = 0; i < 6; i++){
        cout<<"Division "<<i+1<<"  ";
        for (int j = 0; j < 4; j++){
            cout<<setw(5)<<divsales[i].getsales(j)<<" ";
        }
        cout<<"\n---------------------------------------\n";
    }
    int sum = 0;
    for(int i = 0; i < 6 ; i++){
        sum += divsales[i].getsum();
    }
    cout<<"The whole sales is "<<setw(7)<<sum<<endl;
    return 0;
}
```

Output

```
==========division sale==========
-              # 1              -
Sales for season 1: 1
Sales for season 2: 1
Sales for season 3: 1
Sales for season 4: 1
================================
==========division sale==========
-              # 2              -
Sales for season 1: 2
Sales for season 2: 2
Sales for season 3: 2
Sales for season 4: 2
================================
==========division sale==========
-              # 3              -
Sales for season 1: 3
Sales for season 2: 3
Sales for season 3: 3
Sales for season 4: 3
================================
==========division sale==========
-              # 4              -
Sales for season 1: 4
Sales for season 2: 4
Sales for season 3: 4
Sales for season 4: 4
================================
==========division sale==========
-              # 5              -
Sales for season 1: 5
Sales for season 2: 5
Sales for season 3: 5
Sales for season 4: 5
================================
==========division sale==========
-              # 6              -
Sales for season 1: 6
Sales for season 2: 6
Sales for season 3: 6
Sales for season 4: 6
================================

=============Final report=============
Division 1      1     1     1     1
---------------------------------------
Division 2      2     2     2     2
---------------------------------------
Division 3      3     3     3     3
---------------------------------------
Division 4      4     4     4     4
---------------------------------------
Division 5      5     5     5     5
---------------------------------------
Division 6      6     6     6     6
---------------------------------------
The whole sales is 84
```

5.

5.cpp > Rational

```cpp
#include <iostream>
#include <cstdlib>
using namespace std;
class Rational{
    private:
        int num;
        int den;
        void reduce(){
            int gcd = this->gcd(abs(num), abs(den));
            num = num/gcd;
            den = den/gcd;
        }
        int gcd(int x, int y) {
            int r = 0;
            while(y != 0) {
                r = x % y;
                x = y;
                y = r;
            }
            return x;
        }
    public:
    Rational(int num, int den){
        this->num = num;
        this->den = den;
    }

    friend ostream& operator<<(ostream &os, const Rational &r) {
        return os << r.rational_to_string();
    }
    Rational operator+(const Rational &that) {
        Rational rational(
            this->num * that.den + that.num * this->den,
            this->den * that.den
        );
        rational.reduce();
        return rational;
    }
    Rational operator-(const Rational &that) {
        Rational rational(
            this->num * that.den - that.num * this->den,
            this->den * that.den
        );
        rational.reduce();
        return rational;
    }
    Rational operator*(const Rational &that) {
        Rational rational(
            this->num * that.num,
            this->den * that.den
        );
        rational.reduce();
        return rational;
    }
    Rational operator/(const Rational &that) {
```

```
        Rational rational(
            this->num * that.den,
            this->den * that.num
        );
        rational.reduce();
        return rational;
    }
    string rational_to_string() const {
        return to_string(this->num) + "/" + to_string(this->den);
    }

};

int main(int argc, char* argv[]){
    int num;
    int den;
    char option, tmp;
    while(true){
        cout<<"input something:";
        cin>>num>>tmp>>den;
        Rational rational1(num, den);

        cin>>option;
        cin>>num>>tmp>>den;
        Rational rational2(num, den);
        if(option == '+')      cout<<"="<<(rational1 + rational2)<<endl;
        else if(option == '-') cout<<"="<<(rational1 - rational2)<<endl;
        else if(option == '*') cout<<"="<<(rational1 * rational2)<<endl;
        else if(option == '/') cout<<"="<<(rational1 / rational2)<<endl;
    }
    return 0;
}
```

Output

```
input something:2/3+6/9
=4/3
input something:1/7-14/1
=-97/7
input something:3/5*1/2
=3/10
input something:3/5/1/2
=6/5
```