

Que1

```
1  #include <iostream>
2
3  using namespace std;
4
5  class Clock
6  {
7  private:
8      int hour, minute, second;
9
10 public:
11     Clock(int h, int m, int s);
12     void Display();
13     Clock operator+(const Clock &c) const;
14 };
15
16 Clock::Clock(int h, int m, int s)
17 {
18     this->hour = h % 24;
19     this->minute = m % 60;
20     this->second = s % 60;
21 }
22
23 Clock Clock::operator+(const Clock &c) const
24 {
25     Clock NewClock(0, 0, 0);
26     NewClock.hour = ((this->hour + c.hour) + (this->minute + c.minute + (this->second + c.second) / 60) / 60) % 24;
27     NewClock.minute = ((this->minute + c.minute) + (this->second + c.second) / 60) % 60;
28     NewClock.second = (this->second + c.second) % 60;
29     return NewClock;
30 }
31
32 void Clock::Display()
33 {
34     printf("%02d : %02d : %02d\n", hour, minute, second);
35 }
36 int main()
37 {
38     Clock c1(3, 4, 5);
39     cout << "Clock.1 is ";
40     c1.Display();
41
42     Clock c2(12, 40, 55);
43     cout << "Clock.2 is ";
```

```
44     c2.Display();
45
46     cout << "Clock.1 + Clock.2 is "<<endl;
47     (c1 + c2).Display();
48 }
```

Que2

```
1  #include <iostream>
2
3  using namespace std;
4
5  class Person
6  {
7  protected:
8      string name;
9      string gender;
10     string birth;
11     int ID;
12
13 public:
14     virtual void Display();
15     Person(string name, string gender, string birth, int ID);
16     Person(const Person &s)
17     {
18         this->birth = s.birth;
19         this->gender = s.gender;
20         this->name = s.name;
21         this->ID = s.ID;
22     }
23 };
24
25 Person ::Person(string name, string gender, string birth, int ID)
26 {
27     this->name = name;
28     this->gender = gender;
29     this->birth = birth;
30     this->ID = ID;
31 }
32
33 void Person::Display()
34 {
35     cout << name << endl;
36 }
37
38 class Student : public Person
39 {
```

```

40 protected:
41     string native_place;
42     int student_id;
43     int age;
44     int score;
45
46 public:
47     Student(string name, string gender, string birth, int ID, string native_place,
48 int studentid, int age, int score);
49     void Display();
50     Student(const Student &s) : Person(s)
51     {
52         this->name = s.name;
53         this->gender = s.gender;
54         this->birth = s.birth;
55         this->age = s.age;
56         this->ID = s.ID;
57         this->native_place = s.native_place;
58         this->student_id = s.student_id;
59         this->score = s.score;
60     }
61     Student operator+(const Student &s) const;
62 };
63 Student::Student(string name, string gender, string birth, int ID, string
64 native_place, int studentid, int age, int score) : Person(name, gender, birth, ID)
65 {
66     this->native_place = native_place;
67     this->student_id = studentid;
68     this->age = age;
69     this->score = score;
70 }
71 Student Student::operator+(const Student &s) const
72 {
73     int score_sum = this->score + s.score;
74     Student NewStudent("sum", "null", 0, 0, "null", 0, 00, score_sum);
75
76     return NewStudent;
77 }
78
79 void Student::Display()
80 {
81     cout << "Student:" << endl;
82     cout << "Basic info:" << endl;
83     cout << "姓名\t性别\t出生日期\t身份ID\n";
84     cout << this->name << "\t" << this->gender << "\t" << this->birth << "\t" <<
85 this->ID << endl;
86     cout << "籍贯\t学号\t年龄\t成绩\n";

```

```

86     cout << this->native_place << "\t" << this->student_id << "\t" << this->age <<
    "\t" << this->score << endl;
87 }
88
89 class Teacher : public Person
90 {
91     protected:
92         string position;
93
94     public:
95         Teacher(string name, string gender, string birth, int ID, string position);
96         void Display();
97 };
98
99 Teacher::Teacher(string name, string gender, string birth, int ID, string
position) : Person(name, gender, birth, ID)
100 {
101     this->position = position;
102 }
103
104 void Teacher::Display()
105 {
106     cout << "Teacher:" << endl;
107     cout << "Basic info:" << endl;
108     cout << "姓名\t性别\t出生日期\t身份ID\n";
109     cout << this->name << "\t" << this->gender << "\t" << this->birth << "\t" <<
this->ID << endl;
110     cout << "职称" << endl;
111     cout << this->position << endl;
112 }
113
114 class Stu_Teacher : virtual public Student, virtual public Teacher
115 {
116     public:
117         Stu_Teacher(string name, string gender, string birth, int ID, string
native_place, int studentid, int age, int score, string position);
118         void Display();
119 };
120
121 Stu_Teacher::Stu_Teacher(string name, string gender, string birth, int ID, string
native_place, int studentid, int age, int score, string position) : Student(name,
gender, birth, ID, native_place, studentid, age, score), Teacher(name, gender,
birth, ID, position)
122 {
123 }
124
125 void Stu_Teacher::Display()
126 {
127     cout << "Student Teacher" << endl;

```

```
128     // cout << "姓名\t性别\t出生日期\t身份ID\n";
129     // cout << this->name << "\t" << this->gender << "\t" << this->birth << "\t"
    << this->ID << endl;
130     // cout << "籍贯\t学号\t年龄\t成绩\n";
131     // cout << this->native_place << "\t" << this->student_id << "\t" << this->age
    << "\t" << this->score << endl;
132     // cout << "职称" << endl;
133     // cout << this->position << endl;
134 }
135
136 int main()
137 {
138     Person *p;
139     // p.Display();
140
141     Student s1("Hellen", "Female", "1998.05.09", 41223874, "江苏", 73913, 24, 78);
142     p = &s1;
143     p->Display();
144
145     Student s2("Kate", "Female", "1997.06.09", 45523456, "湖北", 76153, 25, 89);
146     p = &s2;
147     p->Display();
148
149     Student s3(s2);
150     s3.Display();
151
152     p = new Teacher("John", "Male", "1970.07.18", 68624053, "Professor");
153     p->Display();
154
155     Stu_Teacher st("Stephen", "Male", "1998.08.09", 65286523, "湖南", 56924, 24,
86, "助教");
156     st.Display();
157
158     (s1 + s2).Display();
159 }
```