

```

1: #include <iostream>
2: #include <fstream>
3: #include <string>
4: #include <iomanip>
5:
6: using namespace std;
7:
8: class Employee {
9: public:
10:     Employee (string name_, double years_of_service_)
11:         : name(name_), years_of_service(years_of_service_) {}
12:     friend ostream & operator<< (ostream &ofs, Employee &rhs);
13:     int get_salary () { return salary; }
14: protected:
15:     string name;
16:     double years_of_service;
17:     int salary;
18: };
19:
20: ostream & operator<< (ostream &ofs, Employee &rhs) {
21:     ofs << left << setw(30) << rhs.name << setw(13)
22:     << rhs.years_of_service << setw(15) << rhs.salary;
23:     return ofs;
24: }
25:
26: class Parttimer : public Employee {
27: public:
28:     Parttimer (string name_, double years_of_service_)
29:         : Employee(name_, years_of_service_)
30:         { salary = 20000 + 1000 * years_of_service; }
31: };
32:
33: class Manager : public Employee {
34: public:
35:     Manager (string name_, double years_of_service_)
36:         : Employee(name_, years_of_service_)
37:         { salary = 35000 + 5000 * years_of_service; }
38: };
39:
40: class Chairman : public Manager {
41: public:
42:     Chairman (string name_, double years_of_service_)
43:         : Manager(name_, years_of_service_)
44:         { salary = salary + 50000; }
45: };
46:

```

```

47: void print_out (Employee **employee_list, int num_employee) {
48:     // selection sort
49:     for (int i=0; i<num_employee; ++i) {
50:         for (int j=i+1; j<num_employee; ++j)
51:             if ( (*employee_list[i]).get_salary() < (*employee_list[j]).get_salary())
52:                 Employee *tmp;
53:                 tmp = employee_list[i];
54:                 employee_list[i] = employee_list[j];
55:                 employee_list[j] = tmp;
56:             }
57:         cout << *employee_list[i] << endl;
58:     }
59: }
60:
61: int main()
62: {
63:     ifstream ifs("input1.txt");
64:     Employee **employee_list;
65:     int num_employee;
66:     ifs >> num_employee;
67:     employee_list = new Employee *[num_employee];
68:     for (int i=0; i<num_employee; ++i) {
69:         string name, title;
70:         double years_of_service;
71:         ifs >> name >> title >> years_of_service;
72:         if (title == "P")
73:             employee_list[i] = new Parttimer(name, years_of_service);
74:         else if (title == "M")
75:             employee_list[i] = new Manager(name, years_of_service);
76:         else
77:             employee_list[i] = new Chairman(name, years_of_service);
78:     }
79:     print_out(employee_list, num_employee);
80:     for (int i=0; i<num_employee; ++i)
81:         delete employee_list[i];
82:     delete[] employee_list;
83:     ifs.close();
84:     return 0;
85: }

```