Problem Statements

1. **University Course Management:** You have a dictionary representing university courses. The keys are course codes, and the values are nested dictionaries containing 'professor', 'students' (a list of student names), and 'schedule' (a nested dictionary with days of the week as keys and time slots as values). Write a program to update the schedule for a particular course, given the course code and new schedule.

```
courses = {
    'CS101': {
        'professor': 'Dr. Smith',
        'students': ['Alice', 'Bob'],
        'schedule': {'Monday': '10:00-11:30', 'Wednesday': '10:00-11:30'}
    },
    'MATH202': {
        'professor': 'Dr. Johnson',
        'students': ['Charlie', 'David'],
        'schedule': {'Tuesday': '12:00-13:30', 'Thursday': '12:00-13:30'}
    },
    'PHYS303': {
        'professor': 'Dr. Lee',
        'students': ['Eve', 'Frank'],
        'schedule': {'Monday': '14:00-15:30', 'Wednesday': '14:00-15:30'}
    }
}
```

2. **Company Hierarchy Management:** You have a dictionary representing a company's hierarchy. The keys are department names, and the values are nested dictionaries containing 'manager', 'employees' (a list of tuples with employee name and position), and 'budget'. Write a program to add a new employee to a specific department.

```
company = {
    'HR': {
        'manager': 'Alice',
        'employees': [('Bob', 'Assistant'), ('Charlie', 'Recruiter')],
        'budget': 50000
    },
    'Engineering': {
        'manager': 'David',
        'employees': [('Eve', 'Developer'), ('Frank', 'Tester')],
        'budget': 150000
    },
    'Sales': {
        'manager': 'Grace',
        'employees': [('Heidi', 'Sales Rep'), ('Ivan', 'Account Manager')],
        'budget': 100000
    }
}
```

3. **Hospital Patient Management System:** You have a dictionary representing patients in a hospital. The keys are patient IDs, and the values are nested dictionaries containing 'name', 'age', 'illness', and 'treatment' (another dictionary with 'doctor' and 'medication'). Write a program to update the treatment information for a specific patient.

```
patients = {
    101: {
        'name': 'Alice',
        'age': 30,
        'illness': 'Flu',
        'treatment': {'doctor': 'Dr. Brown', 'medication': 'Tamiflu'}
    },
    102: {
        'name': 'Bob',
        'age': 45,
        'illness': 'Diabetes',
        'treatment': {'doctor': 'Dr. Green', 'medication': 'Metformin'}
    },
    103: {
        'name': 'Charlie',
        'age': 28,
        'illness': 'Flu',
        'treatment': {'doctor': 'Dr. Brown', 'medication': 'Tamiflu'}
    }
}
```

4. **Library Book Tracking System:** You have a dictionary representing books in a library. The keys are book IDs, and the values are nested dictionaries containing 'title', 'author', 'borrowers' (a list of dictionaries with 'borrower name' and 'borrowed date'). Write a program to add a new borrower to a specific book.

```
library = {
    1001: {
        'title': 'Book1',
        'author': 'Author1',
        'borrowers': [{'name': 'Alice', 'date': '2024-01-15'}, {'name':
'Bob', 'date': '2024-02-10'}]
    },
    1002: {
        'title': 'Book2',
        'author': 'Author2',
        'borrowers': [{'name': 'Charlie', 'date': '2024-03-20'}]
    },
    1003: {
        'title': 'Book3',
        'author': 'Author3',
        'borrowers': []
    }
}
```

5. **City Weather Data Analysis:** You have a dictionary representing weather data for cities. The keys are city names, and the values are nested dictionaries containing 'coordinates' (a tuple with latitude and longitude), 'temperature' (a list of daily temperatures for a month), and 'humidity' (a list of daily humidity percentages for a month). Write a program to calculate the average temperature and humidity for each city.

```
weather = {
    'New York': {
        'coordinates': (40.7128, -74.0060),
        'temperature': [75, 77, 72, 70, 68, 73, 76, 78, 79, 80, 75, 77, 72,
70, 68, 73, 76, 78, 79, 80, 75, 77, 72, 70, 68, 73, 76, 78, 79, 80],
        'humidity': [60, 62, 65, 63, 61, 64, 66, 67, 68, 70, 60, 62, 65, 63,
61, 64, 66, 67, 68, 70, 60, 62, 65, 63, 61, 64, 66, 67, 68, 70]
    },
    'Los Angeles': {
        'coordinates': (34.0522, -118.2437),
        'temperature': [80, 79, 81, 78, 77, 80, 82, 83, 84, 85, 80, 79, 81,
78, 77, 80, 82, 83, 84, 85, 80, 79, 81, 78, 77, 80, 82, 83, 84, 85],
        'humidity': [55, 57, 60, 58, 56, 59, 61, 62, 63, 65, 55, 57, 60, 58,
56, 59, 61, 62, 63, 65, 55, 57, 60, 58, 56, 59, 61, 62, 63, 65]
    }
}
```

6. **Country Demographics Analysis:** You have a dictionary representing demographics data for countries. The keys are country names, and the values are nested dictionaries containing 'population', 'languages' (a list of official languages), and 'cities' (another dictionary with city names as keys and their populations as values). Write a program to find the most populated city in each country.

```
demographics = {
    'USA': {
        'population': 331002651,
        'languages': ['English'],
        'cities': {'New York': 8419000, 'Los Angeles': 3980000, 'Chicago':
2716000}
    },
    'India': {
        'population': 1380004385,
        'languages': ['Hindi', 'English'],
        'cities': {'Mumbai': 12442373, 'Delhi': 11007835, 'Bangalore':
8436675}
    },
    'China': {
        'population': 1439323776,
        'languages': ['Chinese'],
        'cities': {'Shanghai': 24183300, 'Beijing': 21542000, 'Chongqing':
15872000}
    }
}
```

7. **Retail Store Inventory Management:** You have a dictionary representing inventory in a retail store. The keys are product IDs, and the values are nested dictionaries containing 'product name', 'category', 'stock' (a list of dictionaries with 'location' and 'quantity'), and 'price'. Write a program to find the total quantity of a specific product across all locations.

```
inventory = {
    101: {
        'product_name': 'Product1',
        'category': 'Category1',
        'stock': [{'location': 'A1', 'quantity': 50}, {'location': 'B1',
'quantity': 30}],
        'price': 15.0
    },
    102: {
        'product_name': 'Product2',
        'category': 'Category2',
        'stock': [{'location': 'A2', 'quantity': 20}, {'location': 'B2',
'quantity': 40}],
        'price': 20.0
    },
    103: {
        'product_name': 'Product3',
        'category': 'Category3',
        'stock': [{'location': 'A3', 'quantity': 10}, {'location': 'B3',
'quantity': 60}],
        'price': 25.0
    }
}
```

8. **National Park Visitor Statistics:** You have a dictionary representing visitor statistics for national parks. The keys are park names, and the values

are nested dictionaries containing 'location', 'yearly visitors' (a list of tuples with year and number of visitors), and 'facilities' (another dictionary with facility names as keys and their availability status as values). Write a program to find the year with the highest visitors for each park.

```
parks = {
    'Yellowstone': {
        'location': 'Wyoming',
        'yearly_visitors': [(2018, 4100000), (2019, 4200000), (2020, 3500000)],
        'facilities': {'camping': True, 'lodging': True, 'hiking': True}
    },
    'Yosemite': {
        'location': 'California',
        'yearly_visitors': [(2018, 4200000), (2019, 4300000), (2020, 3700000)],
        'facilities': {'camping': True, 'lodging': True, 'hiking': True}
    }
}
```

9. **E-commerce Order Management:** You have a dictionary representing customer orders in an e-commerce platform. The keys are order IDs, and the values are nested dictionaries containing 'customer' (another dictionary with 'name' and 'address'), 'items' (a list of dictionaries with 'item name', 'quantity', and 'price per unit'), and 'status'. Write a program to calculate the total price for each order.

```
orders = {
    1001: {
        'customer': {'name': 'Alice', 'address': '123 Main St'},
        'items': [{'name': 'Item1', 'quantity': 2, 'price': 15.0}, {'name':
'Item2', 'quantity': 1, 'price': 30.0}],
        'status': 'shipped'
    },
    1002: {
        'customer': {'name': 'Bob', 'address': '456 Elm St'},
        'items': [{'name': 'Item3', 'quantity': 1, 'price': 25.0}, {'name':
'Item4', 'quantity': 3, 'price': 10.0}],
        'status': 'processing'
    }
}
```

10. **Restaurant Menu and Orders System:** You have a dictionary representing a restaurant's menu and orders. The keys are menu item IDs, and the values are nested dictionaries containing 'name', 'category', 'price', and 'orders' (a list of dictionaries with 'customer' and 'quantity'). Write a program to calculate the total revenue generated from each menu item.

```
menu = {
    1: {
        'name': 'Burger',
        'category': 'Main Course',
        'price': 5.0,
        'orders': [{'customer': 'Alice', 'quantity': 2}, {'customer': 'Bob',
'quantity': 1}]
    },
    2: {
        'name': 'Fries',
        'category': 'Side Dish',
        'price': 2.0,
        'orders': [{'customer': 'Charlie', 'quantity': 3}, {'customer':
'David', 'quantity': 2}]
    },
    3: {
        'name': 'Coke',
        'category': 'Beverage',
        'price': 1.5,
        'orders': [{'customer': 'Eve', 'quantity': 2}, {'customer': 'Frank',
'quantity': 1}]
    }
}
```