-16

Q1

```
#include <iostream>
using namespace std;
class Numbers {
private:
int num1, num2;
public:
// Default constructor
Numbers() {
num1 = 0;
num2 = 0;
}
// Parameterized constructor with default values
Numbers(int n1, int n2 = 0) {
num1 = n1;
num2 = n2;
}
// Function to display maximum of two numbers
void displayMax() {
cout << "Maximum of " << num1 << " and " << num2 << " is: " << (num1 >
num2 ? num1 : num2) << endl;
}
};
int main() {
// Creating and initializing objects using default constructor and
parameterized constructor with default values
Numbers obj1; // Default constructor
Numbers obj2(5); // Parameterized constructor with default value for num2
Numbers obj3(10, 7); // Parameterized constructor with provided values
// Displaying maximum for all objects
```

```
obj1.displayMax();
obj2.displayMax();
obj3.displayMax();
return 0;
}
```

Q2

```
#include <iostream>
using namespace std;
class Time
{
private:
int hours;
int minutes;
int seconds;
public:
Time() : hours(0), minutes(0), seconds(0) {}
friend istream &operator>>(istream &in, Time &t);
friend ostream &operator<<(ostream &out, const Time &t);
int totalSeconds()
{
return hours * 3600 + minutes * 60 + seconds;
}
};
istream &operator>>(istream &in, Time &t)
{
cout << "Enter hours: ";
in >> t.hours;
cout << "Enter minutes: ";
in >> t.minutes;
cout << "Enter seconds: ";
in >> t.seconds;
```

```
return in;
}
ostream &operator<<(ostream &out, const Time &t)
{
out << t.hours << " hours, " << t.minutes << " minutes, " << t.seconds <<
" seconds";
return out;
}
int main()
{
Time t;
cout << "Enter time:" << endl;
cin >> t;
cout << "Entered time: " << t << endl;
cout << "Total seconds: " << t.totalSeconds() << endl;
return 0;
}
```