

Samir Khadka

CS360 - Programming in C and C++

Assignment 2

1. Answer

```
1  #include <iostream>
2  #include <string>
3
4  class GradeBook {
5  public:
6      explicit GradeBook(std::string, std::string);
7      void setCourseName(std::string);
8      void setInstructorName(std::string);
9      std::string getCourseName() const;
10     std::string getInstructorName() const;
11     void displayMessage() const;
12 private:
13     std::string courseName;
14     std::string instructorName;
15 };
16
17 GradeBook::GradeBook(std::string course, std::string instructor)
18     : courseName(course), instructorName(instructor) {}
19
20 void GradeBook::setCourseName(std::string name) {
21     courseName = name;
22 }
23
24 void GradeBook::setInstructorName(std::string name) {
25     instructorName = name;
26 }
27
28 std::string GradeBook::getCourseName() const {
29     return courseName;
30 }
31
```

```

std::string GradeBook::getInstructorName() const {
    return instructorName;
}

void GradeBook::displayMessage() const {
    std::cout << "Welcome to the grade book for\n" << getCourseName()
               << "!\nThis course is presented by: " << getInstructorName()
               << std::endl;
}

int main() {
    GradeBook myGradeBook("C++ Programming", "Dr. Smith");

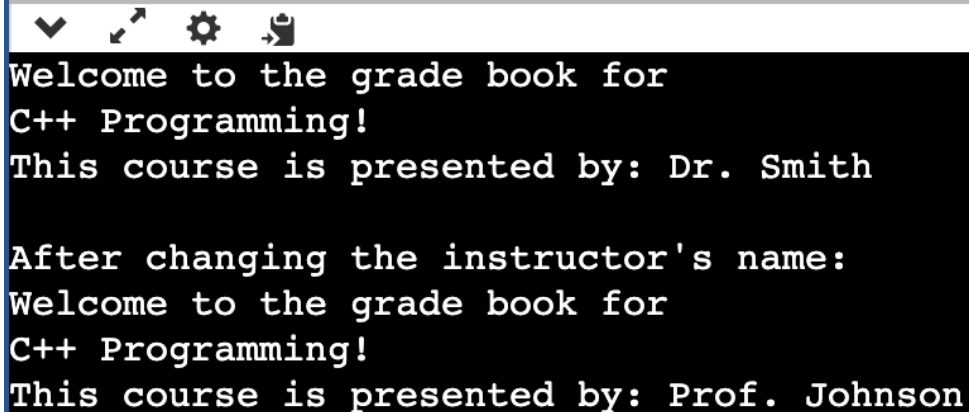
    myGradeBook.displayMessage();

    myGradeBook.setInstructorName("Prof. Johnson");

    std::cout << "\nAfter changing the instructor's name:\n";
    myGradeBook.displayMessage();

    return 0;
}

```



```

Welcome to the grade book for
C++ Programming!
This course is presented by: Dr. Smith

After changing the instructor's name:
Welcome to the grade book for
C++ Programming!
This course is presented by: Prof. Johnson

```

2. Answer:

```
main.cpp
1  #include <iostream>
2
3  class Date {
4  public:
5      Date(int m, int d, int y);
6      void setMonth(int m);
7      void setDay(int d);
8      void setYear(int y);
9      int getMonth() const;
10     int getDay() const;
11     int getYear() const;
12     void displayDate() const;
13 private:
14     int month;
15     int day;
16     int year;
17 };
18
19
20 Date::Date(int m, int d, int y) {
21     setMonth(m);
22     setDay(d);
23     setYear(y);
24 }
25
26
27 void Date::setMonth(int m) {
28     month = (m >= 1 && m <= 12) ? m : 1;
29 }
30
31
```

main.cpp

```
32 void Date::setDay(int d) {
33     day = d;
34 }
35
36
37 void Date::setYear(int y) {
38     year = y;
39 }
40
41 int Date::getMonth() const {
42     return month;
43 }
44
45 int Date::getDay() const {
46     return day;
47 }
48
49 int Date::getYear() const {
50     return year;
51 }
52
53 void Date::displayDate() const {
54     std::cout << month << "/" << day << "/" << year << std::endl;
55 }
56
57 int main() {
58     Date myDate(2, 20, 2024);
59
60     std::cout << "Initial date: ";
61     myDate.displayDate();
62
63     myDate.setMonth(12);
64     myDate.setDay(31);
65     myDate.setYear(2025);
66
67     std::cout << "Updated date: ";
68     myDate.displayDate();
69
70     return 0;
71 }
72
```

Initial date: 2/20/2024  
Updated date: 12/31/2025

### 3. Answer:

```
main.cpp
1  #include <iostream>
2  #include <string>
3
4  class HeartRates {
5  public:
6      HeartRates(std::string fName, std::string lName, int birthMonth, int birthDay, int birthYear);
7
8      void setFirstName(std::string fName);
9      std::string getFirstName() const;
10
11     void setLastName(std::string lName);
12     std::string getLastName() const;
13
14     void setBirthMonth(int month);
15     int getBirthMonth() const;
16
17     void setBirthDay(int day);
18     int getBirthDay() const;
19
20     void setBirthYear(int year);
21     int getBirthYear() const;
22
23     int getAge(int currentMonth, int currentDay, int currentYear) const;
24     int getMaximumHeartRate(int currentMonth, int currentDay, int currentYear) const;
25     std::pair<int, int> getTargetHeartRate(int currentMonth, int currentDay, int currentYear) const;
26
27 private:
28     std::string firstName;
29     std::string lastName;
30     int birthMonth;
31     int birthDay;
32     int birthYear;
33 };
34
35 HeartRates::HeartRates(std::string fName, std::string lName, int birthMonth, int birthDay, int birthYear)
36     : firstName(fName), lastName(lName), birthMonth(birthMonth), birthDay(birthDay), birthYear(birthYear) {}
37
38 void HeartRates::setFirstName(std::string fName) {
39     firstName = fName;
40 }
41
42 std::string HeartRates::getFirstName() const {
43     return firstName;
44 }
45
46 void HeartRates::setLastName(std::string lName) {
47     lastName = lName;
48 }
49
50 std::string HeartRates::getLastName() const {
51     return lastName;
52 }
53
54 void HeartRates::setBirthMonth(int month) {
55     birthMonth = month;
56 }
57
58 int HeartRates::getBirthMonth() const {
59     return birthMonth;
60 }
61
62 void HeartRates::setBirthDay(int day) {
```

main.cpp

```
63     birthDay = day;
64 }
65
66 int HeartRates::getBirthDay() const {
67     return birthDay;
68 }
69
70 void HeartRates::setBirthYear(int year) {
71     birthYear = year;
72 }
73
74 int HeartRates::getBirthYear() const {
75     return birthYear;
76 }
77
78 int HeartRates::getAge(int currentMonth, int currentDay, int currentYear) const {
79     int age = currentYear - birthYear;
80     if (currentMonth < birthMonth || (currentMonth == birthMonth && currentDay < birthDay)) {
81         age--;
82     }
83     return age;
84 }
85
86 int HeartRates::getMaximumHeartRate(int currentMonth, int currentDay, int currentYear) const {
87     return 220 - getAge(currentMonth, currentDay, currentYear);
88 }
89
90 std::pair<int, int> HeartRates::getTargetHeartRate(int currentMonth, int currentDay, int currentYear) const {
91     int maxHeartRate = getMaximumHeartRate(currentMonth, currentDay, currentYear);
92     int minTarget = maxHeartRate * 0.5;
93     int maxTarget = maxHeartRate * 0.85;
```

main.cpp

```
94     return std::make_pair(minTarget, maxTarget);
95 }
96
97 int main() {
98     std::string firstName, lastName;
99     int birthMonth, birthDay, birthYear;
100
101     std::cout << "Enter first name: ";
102     std::cin >> firstName;
103
104     std::cout << "Enter last name: ";
105     std::cin >> lastName;
106
107     std::cout << "Enter birth month (1-12): ";
108     std::cin >> birthMonth;
109
110     std::cout << "Enter birth day: ";
111     std::cin >> birthDay;
112
113     std::cout << "Enter birth year: ";
114     std::cin >> birthYear;
115
116     HeartRates person(firstName, lastName, birthMonth, birthDay, birthYear);
117
118     int currentMonth, currentDay, currentYear;
119     std::cout << "Enter current month (1-12): ";
120     std::cin >> currentMonth;
121
122     std::cout << "Enter current day: ";
123     std::cin >> currentDay;
124
```



```

111     std::cin >> birthDay;
112
113     std::cout << "Enter birth year: ";
114     std::cin >> birthYear;
115
116     HeartRates person(firstName, lastName, birthMonth, birthDay, birthYear);
117
118     int currentMonth, currentDay, currentYear;
119     std::cout << "Enter current month (1-12): ";
120     std::cin >> currentMonth;
121
122     std::cout << "Enter current day: ";
123     std::cin >> currentDay;
124
125     std::cout << "Enter current year: ";
126     std::cin >> currentYear;
127
128     std::cout << "\nInformation:\n";
129     std::cout << "Name: " << person.getFirstName() << " " << person.getLastName() << std::endl;
130     std::cout << "Date of Birth: " << person.getBirthMonth() << "/" << person.getBirthDay() << "/" << person.getBirthYear() << std::endl;
131     std::cout << "Age: " << person.getAge(currentMonth, currentDay, currentYear) << " years" << std::endl;
132
133     int maxHeartRate = person.getMaximumHeartRate(currentMonth, currentDay, currentYear);
134     std::cout << "Maximum Heart Rate: " << maxHeartRate << " bpm" << std::endl;
135
136     auto targetHeartRate = person.getTargetHeartRate(currentMonth, currentDay, currentYear);
137     std::cout << "Target Heart Rate Range: " << targetHeartRate.first << " bpm - " << targetHeartRate.second << " bpm" << std::endl;
138
139     return 0;
140 }
141

```

Enter first name: Samir  
 Enter last name: Khadka  
 Enter birth month (1-12): 11  
 Enter birth day: 17  
 Enter birth year: 2002  
 Enter current month (1-12): 02  
 Enter current day: 20  
 Enter current year: 2024

Information:  
 Name: Samir Khadka  
 Date of Birth: 11/17/2002

The screenshot shows the OnlineGDB web interface. The left sidebar contains navigation links like 'Question 1 Assign 2', 'Create New Project', 'My Projects', 'Classroom', 'Learn Programming', 'Programming Questions', 'Sign Up', 'Logout', and 'Learn Python with KodeKloud'. The main editor displays a C++ program in 'main.cpp' with line numbers 111 to 141. The code defines a 'HeartRates' struct, prompts for birth date, current date, and current year, then calculates and displays the maximum heart rate and target heart rate range. The 'Run' button is highlighted in the toolbar. Below the code editor, the 'Input' section shows the user's input '2024' for the current year. The 'Output' section displays the program's results: Name: Samir Khadka, Date of Birth: 11/17/2002, Age: 21 years, Maximum Heart Rate: 199 bpm, and Target Heart Rate Range: 99 bpm - 169 bpm. The footer includes links for 'About', 'FAQ', 'Blog', 'Terms of Use', 'Contact Us', 'GDB', 'Tutorial', 'Credits', 'Privacy', and a copyright notice for 2010-2024 GDB Online.

```
111 std::cin >> birthDay;
112
113 std::cout << "Enter birth year: ";
114 std::cin >> birthYear;
115
116 HeartRates person(firstName, lastName, birthMonth, birthDay, birthYear);
117
118 int currentMonth, currentDay, currentYear;
119 std::cout << "Enter current month (1-12): ";
120 std::cin >> currentMonth;
121
122 std::cout << "Enter current day: ";
123 std::cin >> currentDay;
124
125 std::cout << "Enter current year: ";
126 std::cin >> currentYear;
127
128 std::cout << "\nInformation:\n";
129 std::cout << "Name: " << person.getFirstName() << " " << person.getLastName() << std::endl;
130 std::cout << "Date of Birth: " << person.getBirthMonth() << "/" << person.getBirthDay() << "/" << person.getBirthYear() << std::endl;
131 std::cout << "Age: " << person.getAge(currentMonth, currentDay, currentYear) << " years" << std::endl;
132
133 int maxHeartRate = person.getMaximumHeartRate(currentMonth, currentDay, currentYear);
134 std::cout << "Maximum Heart Rate: " << maxHeartRate << " bpm" << std::endl;
135
136 auto targetHeartRate = person.getTargetHeartRate(currentMonth, currentDay, currentYear);
137 std::cout << "Target Heart Rate Range: " << targetHeartRate.first << " bpm - " << targetHeartRate.second << " bpm" << std::endl;
138
139 return 0;
140 }
141
```

Enter current year: 2024

Information:  
Name: Samir Khadka  
Date of Birth: 11/17/2002  
Age: 21 years  
Maximum Heart Rate: 199 bpm  
Target Heart Rate Range: 99 bpm - 169 bpm

...Program finished with exit code 0  
Press ENTER to exit console.[]

```
Enter current year: 2024

Information:
Name: Samir Khadka
Date of Birth: 11/17/2002
Age: 21 years
Maximum Heart Rate: 199 bpm
Target Heart Rate Range: 99 bpm - 169 bpm
```