F18 CS20A Quiz1 In Class

STEWART DULANEY

TOTAL POINTS

17 / 19

QUESTION 1

8 pts

1.1 1 / 1

✓ **- 0 pts Correct**

- 1 pts Incorrect

1.2 1 / 1

✓ **- 0 pts Correct**

- 1 pts Incorrect

1.3 1 / 1

✓ **- 0 pts Correct**

- 1 pts incorrect

1.4 1 / 1

✓ **- 0 pts Correct**

- 1 pts Incorrect

1.5 1 / 1

✓ **- 0 pts Correct**

- 1 pts Incorrect

1.6 1 / 1

✓ **- 0 pts Correct**

- 1 pts Incorrect

1.7 2 / 2

✓ **- 0 pts Correct**

- 1 pts Incomplete

- 2 pts Incorrect

QUESTION 2

11 pts

2.1 1 / 1

✓ **- 0 pts Correct**

- 1 pts Incomplete/Incorrect

2.2 2 / 2

✓ **- 0 pts Correct**

- 1 pts Click here to replace this description.

- 2 pts Incorrect/Incomplete

2.3 2 / 2

✓ **- 0 pts Correct**

- 1 pts Click here to replace this description.

- 2 pts Incorrect/Incomplete

2.4 2 / 2

✓ **- 0 pts Correct**

- 1 pts Click here to replace this description.

- 2 pts Incorrect/Incomplete

2.5 1 / 2

- 0 pts Correct

✓ **- 1 pts Click here to replace this description.**

- 2 pts Incorrect/Incomplete

2.6 1 / 2

- 0 pts Correct

✓ **- 1 pts Click here to replace this description.**

- 2 pts Incorrect/Incomplete

Name: Stewart Dulaney ID: 1545566

Problem 1:

a) Consider the following program which consists of four functions: main, printArray, swap1 and swap2. For each of the printArray function calls indicate the state of the array at the time of those calls in the boxes to the right of main.

```
void swap1(int* a, int *b) {
      int* temp = a;
      a = b;
      b = temp;
}
```

```
void swap2(int* a, int *b) {
      int temp = *a;
      *a = *b;
      *b = temp;
}
```

```
int main() {

      const int size = 6;
      int arr[size]
            = { 7, 64, 4, 6, 21, 32 };

      int *ptr = arr + 2;
      printArray(arr, size);

      ptr[1] = 11;
      printArray(arr, size);

      ptr += 2;
      printArray(arr, size);

      *ptr = -1;
      printArray(arr, size);

      *(arr + 2) = 81;
      printArray(arr, size);

      swap1(&arr[0], &arr[1]);
      printArray(arr, size);

      swap2( (ptr - 1) , &arr[2]);
      printArray(arr, size);
}
```

```
void printArray(int arr[], int size) {
      for (int i = 0; i < size; i++)
            cout << arr[i] << " ";
      cout << endl;
}
```

7	64	4	6	21	32

7	64	4	11	21	32

7	64	4	11	21	32

7	64	4	11	-1	32

7	64	81	11	-1	32

7	64	81	11	-1	32

7	64	11	81	-1	32

b) In the previous part, one of the swap functions did not swap the values as intended. Discuss which one is incorrect and why?

swap1 is incorrect b/c the only variables changed in the function are local variables. In order to swap the values as intended, the parameters a and b should be dereferenced so that the values they point to can be swapped. Instead, swap1 swaps the addresses contained in a and b, which leaves the values they point to unchanged.

Name: Stewart Dulaney ID: 1545566

Problem 2:

Implement the class Tiktok discussed below. You may assume both iostream and string libraries are included, you may not use any other libraries. There are 60 seconds in a minute, 3600 seconds in an hour, 60 minutes in an hour. The 24-hour clock convention goes from 0:0:0 to 23:59:59. **Read everything** before starting that way you get a better sense on what member variables you need and how you want to implement the member functions.

It may be useful to recall that the modulus operator "%" can be used to obtain the remainder of integer division, also the "+" operator may be used to concatenate strings, and the function std::to_string(int) takes an integer and returns the string equivalent of that integer, presumably for concatenating a string with an integer.

- Implement the default constructor which sets the initial state of Tiktok to correspond to time 0:00:00 in 24-hour clock convention (or 12:00:00 AM 12-hour clock convention).
- addSeconds adds the number of seconds passed into the function to Tiktok. If the amount of seconds to add is negative do nothing. There is no upper limit to the amount of seconds that can be added.
- addMinutes adds the number of minutes passed into the function to Tiktok. If the amount of minutes to add is negative do nothing. There is no upper limit to the amount of minutes that can be added.
- addHours adds the number of hours passed into the function to Tiktok. If the amount of hours to add is negative do nothing. There is no upper limit to the amount of hours that can be added.
- display24 prints to the console the time stored in Tiktok using 24-hour convention; with the following format: XX:XX:XX, followed by a newline.
- display12 prints to the console the time stored in Tiktok using 12-hour convention. It will print AM or PM appropriately; with the following format XX:XX:XX XM, followed by a newline

Below is the class definition for Tiktok, presumably in some header file. After planning out your design declare your member variables in the private field of the class.

Tiktok.h

```
class Tiktok {
public:
      Tiktok();

      void addSeconds(int s);
      void addMinutes(int m);
      void addHours(int h);

      void display24() const;
      void display12() const;

private:
      //TODO: Declare any member variables you think you will need.
      int m_seconds;

};
```

Tiktok.cpp

Santa Monica College CS 20A: Data Structures with C++ Quiz 1
Fall 2018

include "Tiktok.h"
include <iostream> # include <string>

Name: Stewart Dulaney **ID:** 1545566

Implement the member functions of Tiktok below, assume this being done in a separate source file. Be as neat and syntactically correct as possible.

a) Constructor:

```
Tiktok :: Tiktok () {
    m_seconds = 0;
}
```

b) addSeconds:

```
void Tiktok :: addSeconds ( int s ) {
    if ( s < 0 ) { return; }
    m_seconds += s;
}
```

Name: Stewart Dulaney ID: 1545566

c) addMinutes:

```
void Tiktok::addMinutes(int m) {
    if ( m < 0 ) {      return;      }
    m_seconds  +=  (60 * m);
```

}

d) addHours:

```
void Tiktok::addHours(int h) {
    if ( h < 0 ) {   return;      }
    m_seconds  +=  (3600 * h);
```

}

Name: Stewart Delaney **ID:** 1545566

e) display24:

```
void Tiktok::display24() const {
    int h = m_seconds / 3600;
    int m = (m_seconds % 3600) / 60;
    int s = (m_seconds % 3600) % 60;

    std::cout << std::to_string(h) + ":" + std::to_string(m) + ":" +
        std::to_string(s) << std::endl;
}
```

f) display12:

```
void Tiktok::display12() const {
    std::string suffix = "AM";
    int h = m_seconds / 3600;
    if ( h > 11 ) {
        suffix = "PM";
    }
    h = h % 12;
    if ( h == 0 ) {
        h += 12;
    }
    int m = (m_seconds % 3600) / 60;
    int s = (m_seconds % 3600) % 60;
    std::cout << std::to_string(h) + ":" + std::to_string(m) + ":" +
        std::to_string(s) + " " + suffix << std::endl;
```