Midterm Exam 1

Due Jun 30, 2021 at 12:01am **Points** 100 **Questions** 17
Available Jun 29, 2021 at 12:01am - Jun 30, 2021 at 12:01am 24 hours
Time Limit 75 Minutes

Instructions

Hi everyone,

Today we have the first online midterm exam for this course. This is the open book/note exam so you can freely use any provided materials on Canvas. It will be available on Canvas from **00:01 am June 29th to 00:01 am June 30th CST** (24 hours, **U.S. Central Time**) so you can take it anytime during this period but as early as possible to avoid any potential issues (e.g., internet access). Moreover, we have **75 minutes** with only **ONE attempt** to complete and turn in our answers so please make sure you complete all questions before submitting. If there are any problems, please feel free to let me know by email!

Good luck!

Po-wei Harn

This quiz is no longer available as the course has been concluded.

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	69 minutes	100 out of 100

Correct answers are no longer available.

Score for this quiz: **100** out of 100
Submitted Jun 29, 2021 at 5:49pm
This attempt took 69 minutes.

Question 1 **3 / 3 pts**

Which of the following statements about “call by value” and “call by reference” is incorrect?

- “call by value” and “call by reference” are two different ways of passing (or calling) data to functions.
- If data passed to a function using “call by value”, then the value is only changed in the variable used inside the function.
- If data is passed by reference (i.e., call by reference), a memory address of the data is copied to the function. instead of the actual variable as is done in a call by value.
- **In the call-by-value case, if the value of the data passed from a caller to a function is changed in the function, the value is also changed in the caller.**

Question 2

3 / 3 pts

Which control construct repeats a sequence of statements zero or more times?

- **while statement**
- do-while statement
- if-elseif-else statement
- switch statement
- if-else statement

Question 3

3 / 3 pts

Which of the following is not name of a C++ library function?

- abs
- sqrt
- random
- floor

Question 4

3 / 3 pts

Consider the array declaration:

int x[20];

There is no memory allocated for data item x[20].

- True
- False

Question 5

3 / 3 pts

In a do-while loop, the **boolean expression** is executed before each execution of the loop body.

- True
- False

Question 6

3 / 3 pts

Which of these array definitions will NOT set all the indexed variables to 0?

- int array[5];
- int array[5] = {0};
- int array[5] = {0,0};
- int array[5] = {0,0,0};
- int array[5] = {0,0,0,0,0};

Question 7

5 / 5 pts

Rewrite the following **do-while** loop using **for** loop.

```
01.  int n = 100;
02.  do {
03.      cout << "X";
04.      n = n + 100;
05.  }
06.  while (n < 1000);
```

Your Answer:

```
for (int n = 100; n < 1000; n = n + 100) {
                                    cout << "X";
                                                        }
```

Question 8 5 / 5 pts

Rewrite the following **printColor()** function using switch statement.

```
01. enum Colors {
02.     COLOR_BLACK,
03.     COLOR_WHITE,
04.     COLOR_RED,
05. };
06.
07. void printColor(Colors color) {
08.     if (color == COLOR_BLACK)
09.         cout << "Black";
10.     else if (color == COLOR_WHITE)
11.         cout << "White";
12.     else if (color == COLOR_RED)
13.         cout << "Red";
14.     else
15.         cout << "Unknown";
16. }
```

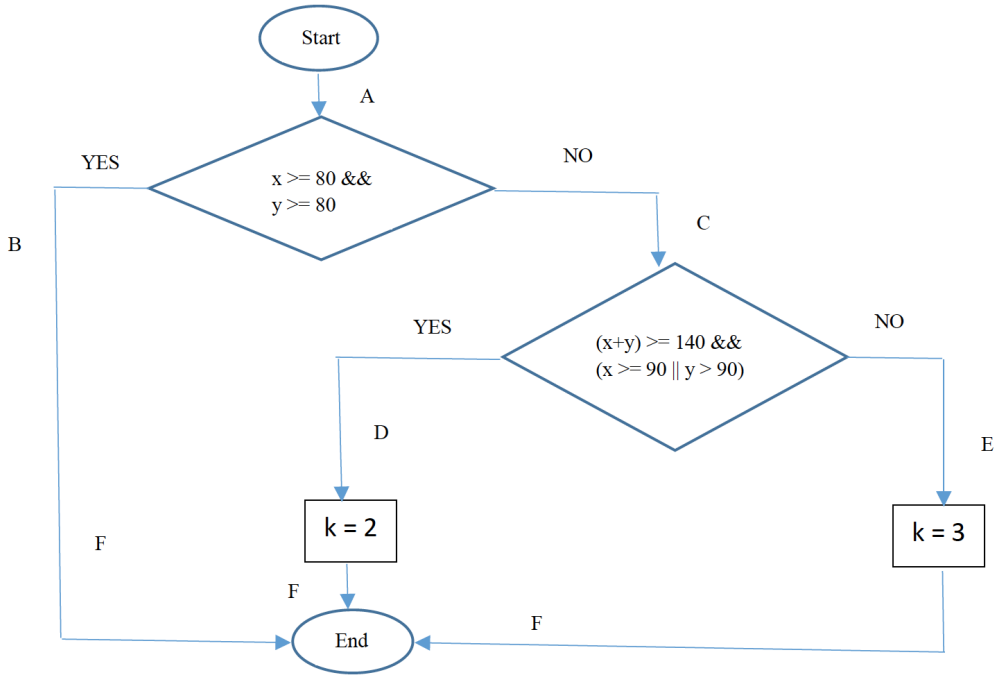
Your Answer:

```
void printColor(Colors color) {
    switch (color) {
        case COLOR_BLACK:
            cout << "Black";
            break;
        case COLOR_WHITE:
            cout << "White";
            break;
        case COLOR_RED:
            cout << "Red";
            break;
        default:
            cout << "Unknown";
            break;
```

}

Please develop test cases (i.e., a pair of values of x and y) to implement the following six coverages.

Use this flow chart for questions **Q9 - Q14**.



Question 9

4 / 4 pts

Statement coverage: Every possible statement in the code should be tested at least once.

Please develop test cases for the **statement coverage**.

Your Answer:

Case 1: x = 200, y = 70 (Covers statements A C D F)

Case 2: x = 50, y = 50 (Covers statements A C E F)

Question 10

6 / 6 pts

Decision coverage: The true or false outcome of each decision point should be tested at least once.

Please develop test cases for the **decision coverage**.

Your Answer:

1. Decision A True: x = 200, y = 200
2. Decision A False: x = 0, y = 0
3. Decision C False: x = 1, y = 1
4. Decision C True: x = 100, y = 70

Question 11

4 / 4 pts

Condition coverage: The true or false outcome of each Boolean expression should be tested at least once.

Please develop test cases for the **condition coverage**.

Your Answer:

1. A: x = 200, y = 200 (T && T)
2. A: x = 0, y = 0 (F && F)
3. C: x = 0, y = 0 (F && (F || F))
4. C: x = 200, y = 0 (T && (T || F))
5. C: x = 0, y = 200 (T && (F || T))

Question 12

8 / 8 pts

Branch coverage: Decision coverage + Condition coverage**.**

Please develop test cases for the **branch coverage**.

Your Answer:

1. A: x = 300, y = 100 (T && T)-true

2. A: x = 5, y = 5 (F && F)-false

3. (A is false) C: x = 5, y= 5 (F && (F || F))-false

4. (A is false) C: x = 300, y = 5 (T && (T || F))-true

5. (A is false) C: x = 5, y = 300 (T && (F || T))-true

Question 13

14 / 14 pts

Branch condition combination coverage: All possibilities of branch condition combination should be tested at least once.

Please develop test cases for the **branch condition combination coverage**.

Your Answer:

1. x = 200, y = 200 (A: T && T)

2. x = 40, y = 200 (A: F && T, C: T && (F || T))

3. x = 70, y = 90 (A: F && T, C: T && (F || F))

4. x = 2, y = 2 (A: F && F, C: F && (F || F))

5. x = 0, y = 82 (A: F && T, C: F && (F || F))

6. x = 200, y = 40 (A: T && F, C: T && (T || F))

7. x = 82, y = 0 (A: T && F, C: F && (F || F))

8. x = 89, y = 70 (A: T && F, C: T && (F || F))

9

Question 14

6 / 6 pts

Path coverage: All paths should be tested at least once.

Please develop test cases for **path coverage**.

Your Answer:

Path 1: A B F (x = 300, y = 400)

Path 2: A C E F (x = 3, y = 3)

Path 3: A C D F (x = 300, y = 30)

Black box test

A U.S. phone number consists of three sections:

1. **Area code**: It can be null or consists of three digits.
2. **Prefix code**: It consists of three digits with non-ZERO or non-ONE as the first digit in the Prefix code section. i.e. 334-123-4567 is invalid because the first digit in the Prefix code is ONE.
3. **Postfix code**: It consists of four digits.

Please list all possible test cases to expose vulnerabilities for the following three tests from questions **Q15 - Q17**.

(**Hints**: Please consider non-numeric values, such as characters)

Question 15

6 / 6 pts

Area code: It can be null or consists of three digits.

Area code test: Please list all possible test cases to expose vulnerabilities of this test.

Your Answer:

1. Test for less than three digits
2. Test for more than three digits
3. Test it when it is null or empty (valid)
4. Test for non-digits like letters, symbols, etc...

Question 16

18 / 18 pts

Prefix code: It consists of three digits with non-ZERO or non-ONE as the first digit in the Prefix code section. i.e. 334-123-4567 is invalid because the first digit in the Prefix code is ONE.

Prefix code test: Please list all possible test cases to expose vulnerabilities of this test.

Your Answer:

1. Test with three digits and the first is not 1 or 0 (valid)
2. Test for three digits starting with 1
3. Test for three digits starting with 0

4. Test for more than three digits

5. Test for less than three digits

6. Test for non-digits like letters, symbols etc...

7. Test if empty

Question 17

6 / 6 pts

Postfix code: It consists of four digits.

Postfix code test: Please list all possible test cases to expose vulnerabilities of this test.

Your Answer:

1. Test if four digits (valid)

2. Test more than four digits

3. Test less than four digits

4. Test for non-digits like letters or symbols

5. Test if empty

Quiz Score: **100** out of 100