Midterm 2 Study Guide

Due No due date Points 25 Questions 25 Time Limit 30 Minutes Allowed Attempts Unlimited

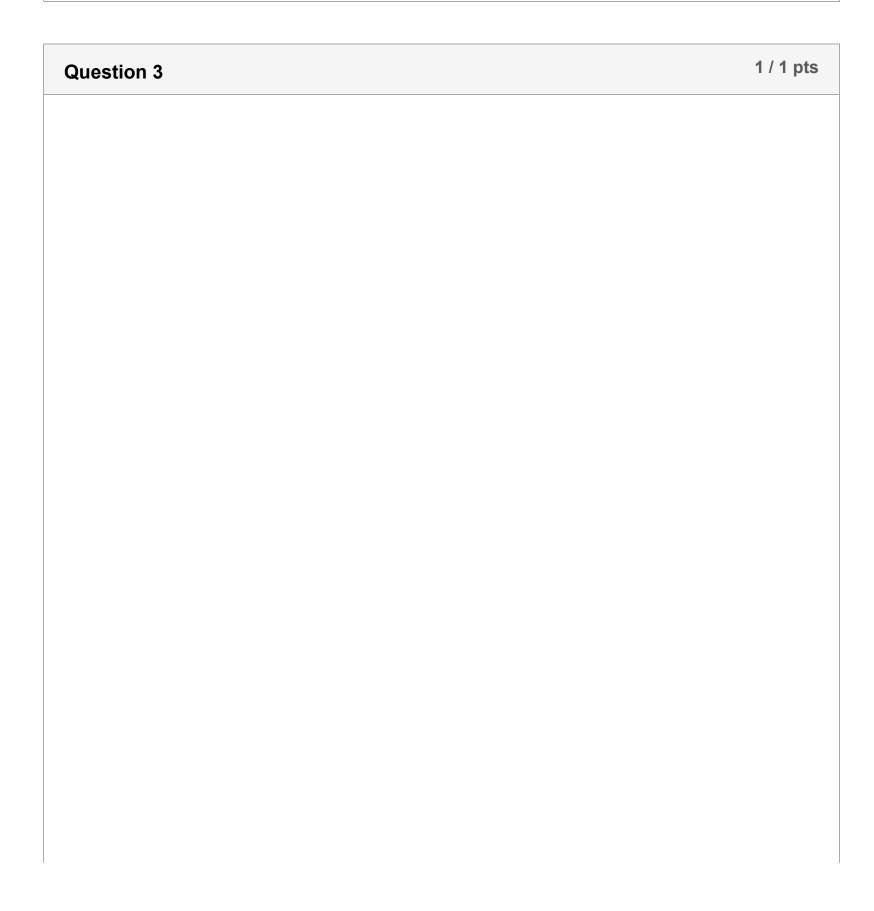
Take the Quiz Again

Attempt History

	Attempt	Time	Score	
KEPT	Attempt 11	29 minutes	25 out of 25	
LATEST	Attempt 19	22 minutes	20 out of 25	
	Attempt 18	30 minutes	23 out of 25	
	Attempt 17	19 minutes	24 out of 25	
	Attempt 16	21 minutes	24 out of 25	
	Attempt 15	30 minutes	23 out of 25	
	Attempt 14	22 minutes	24 out of 25	
	Attempt 13	25 minutes	24 out of 25	
	Attempt 12	30 minutes	22 out of 25	
	Attempt 11	29 minutes	25 out of 25	
	Attempt 10	30 minutes	20.17 out of 25	
	Attempt 9	29 minutes	20 out of 25	
	Attempt 8	29 minutes	20.5 out of 25	
	Attempt 7	28 minutes	21 out of 25	
	Attempt 6	24 minutes	21 out of 25	
	Attempt 5	20 minutes	17.83 out of 25	
	Attempt 4	30 minutes	17.67 out of 25	
	Attempt 3	22 minutes	16 out of 25	
	Attempt 2	16 minutes	16.17 out of 25	
	Attempt 1	30 minutes	18.33 out of 25	

(!) Correct answers are hidden.

Submitted Jun 29 at 4:23pm





Below is the illustration from the loop building strategy. The *highlighted lines* represent. Store the next character from str in current-character:

```
Store the next character from str in current-character:
    Given: the variable str is a string (may be empty)
   Create the counter variable, initialized to {	ext{-}}1
   If the variable str has any characters then
        Set counter to 0
        Create the variable current-character as a character
        Place the first character in str into current-character
        While more-characters and current-character not a period
           Add one to (or increment) the counter variable
          Store the next character from str in current-character
        If current-character is a period then
           Add one to the counter to account for the period.
        Else
          Set counter to -2
   If counter is -1 the string was empty
   Else if counter is -2 there was no period
   goal operation
   goal precondition
   loop bounds
   bounds precondition
   advancing the loop
   loop postcondition
```

False

Question 5 1 / 1 pts

Below is the illustration from the loop building strategy. The *highlighted lines* represent. Create the variable current-chracter as a character:

```
Given: the variable str is a string (may be empty)
Create the counter variable, initialized to -1
If the variable str has any characters then
    Set counter to 0
    Create the variable current-character as a character
    Place the first character in str into current-character
    While more-characters and current-character not a period
       Add one to (or increment) the counter variable
       Store the next character from str in current-character
    If current-character is a period then
       Add one to the counter to account for the period.
      Set counter to -2
If counter is -1 the string was empty
Else if counter is -2 there was no period
advancing the loop
loop bounds
goal precondition
bounds precondition
loop postcondition
goal operation
```

```
Question 7

How many times is this loop entered? (That is, how many times is i printed?)

for (int i = 0; i < 10; i++)
        cout << i;
    cout << endl;
```

O Never			
10			
O 9			
O 11			

Question 8		1 / 1 pts
Match each item with the correct statement belo	DW.	
Actions that occur after the loop is complete	postcondition	~
Actions occurring inside the loop's body	operation	~
Actions that occur before the loop is encountered	precondition	~
A test the determines if the loop should be entered	bounds	~

Question 9	1 / 1 pts
Which line <i>advances the loop</i> ?	
<pre>1. string s("Hello CS 150"); 2. while (s.size()) 3. { 4. if (s.at(0) == 'C') break; 5. s = s.substr(1); 6. } 7. cout << s << endl;</pre>	
O None of these	
O 2	
O 4	
5	

Question 10 The compiler determines which everlanded function to call by leaking at the number, types

The compiler determines which overloaded function to call by looking at the number, types and order of the arguments passed to the function.



True			
O False			

Q	Question 11	1 / 1 pts
V	Which of these documentation tags are used in a <i>function comment?</i>	
	☑ @param	
	□ @file	
	☑ @code	
	<pre>@version</pre>	



Incorrect Question 13 0 / 1 pts

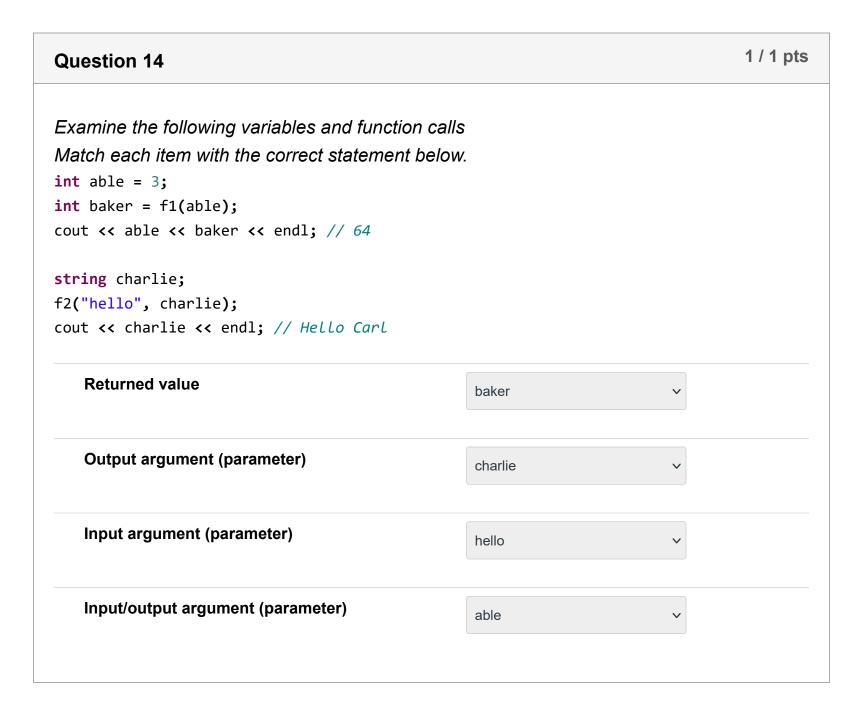
To allow f() to change the argument passed here, the parameter str should be declared as:

```
void f( . . . str);
int main()
{
    f("hello");
}
```

- string&
- O const string&



0	const string
0	It is not possible for $f()$ to change the argument passed here.
0	string



Question 15

Given the overloaded functions prototypes and the variable definition below, which of the function calls will fail to compile?

int f(int&);
int f(int);
int f(int, int);
int a = 7;

| f(a);
| Mone of these fail to compile
| f('a', 'b')
| f(3)



Incorrect

```
1 / 1 pts
Question 16
What prints?
void fn(int, double, double&) { cout << "A" << endl; }</pre>
void fn(int, int, double&) { cout << "B" << endl; }</pre>
void fn(int, int, double) { cout << "C" << endl; }</pre>
void fn(int, int, int) { cout << "D" << endl; }</pre>
int main()
{
    fn(2.5, 1.5, 2.5);
}
   C
   \bigcirc D
   ○ B
   \circ A
   Syntax error: no candidates
   O Syntax error: ambiguous
```

Question 17	1 / 1 pts
An incomplete, yet compilable, linkable and executable function is called a	?
Oprototype	
O declaration	
stub	
O None of these	

Question 18	1 / 1 pts
Unformatted I/O means that you read and write data line-by-line.	
O True	
False	

Question 19 1 / 1 pts

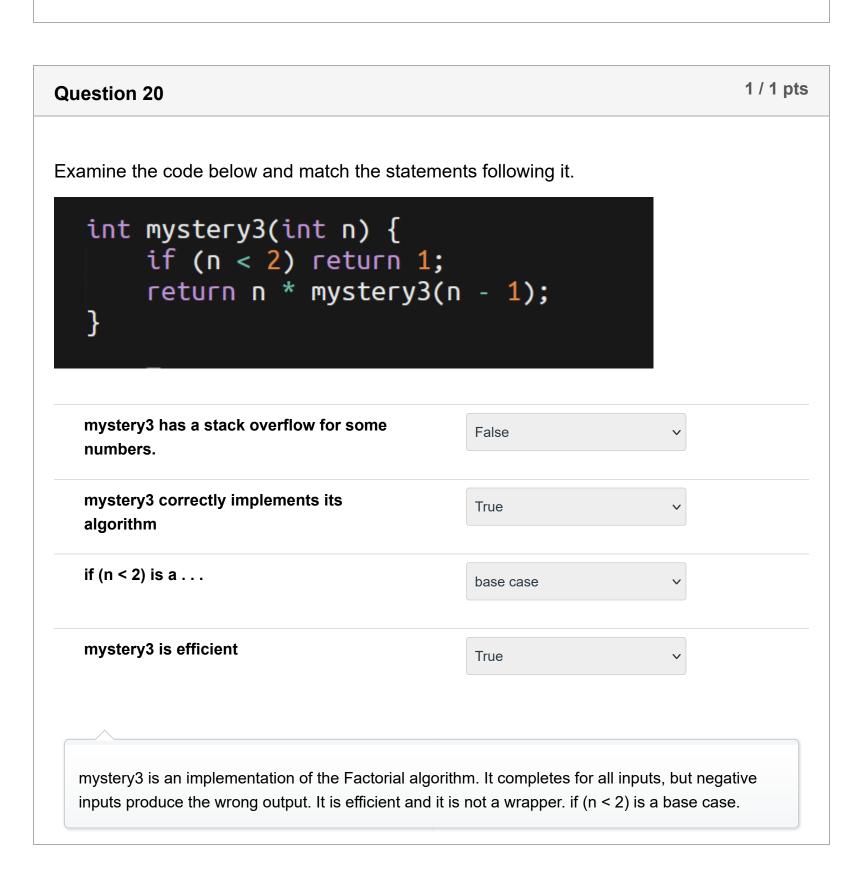
Assuming that you need to write a recursive function calc_prod(int n) to calculate the product of the first n integers, which of the following would be a correct way to simplify the input for the recursive call?

O Call calc_prod(n - 2) and multiply by n.

O Call calc_prod(1) and multiply by n.

O Call calc_prod(n - 1) and multiply by n.

Call calc_prod(n + 1) and multiply by n.





illustrates line-based stream processing	
illustrates token-based stream processing	
illustrates raw character I/O	
O has a syntax error	
is an endless loop	

Question 22	1 / 1 pts
Which line runs the dwk program and gets its input from a file named y.data?	
<pre>./dwk < y.data</pre>	
O None of these	
○ ./dwk >> y.data	
○ ./dwk > y.data	
○ ./dwk y.data	
○ ./dwk << y.data	

```
At the lowest level, all input and output is a stream of bytes flowing through your program.

True

False
```

```
Incorrect Question 24 0 / 1 pts
```

```
What is the value of r("axxbxx")?

string r(const string& s)
{
    auto front = s.substr(0, 1);
    if (front.empty()) return "";
    return (front == "x" ? front : "") + r(s.substr(1));
}

O Stack overflow

O "xxxxx"
```



"ab"			
○ "ax bx "			
O "a b "			

Question 25	1 / 1 pts
If an output stream's file is missing when you try to open it, its fail() member function returns false.	
True	
O False	

