Started on	Saturday, October 1, 2022, 7:55 PM
State	Finished
Completed on	Saturday, October 1, 2022, 9:13 PM
Time taken	1 hour 18 mins

Question 1

Correct

Points out of 1.00

Suppose **w**, **x**, and **y** are variables defined as type **int**. Is the statement

```
1   w = x++ - y;
2
```

equivalent to the following pair of statements?

```
1   x += 1;
2   w = x - y;
3
```

Select one:

○ True

◉ False ✔

Question 2

Correct

Points out of 1.00

Suppose **w**, **x**, and **y** are variables defined as type **int**. Is the statement

```
1   w = ++x - y;
2
```

equivalent to the following pair of statements?

```
1   x = x + 1;
2   w = x - y;
3
```

Select one:

◉ True ✔

○ False

Question 3

Correct

Points out of 1.00

Walk through the following code fragment and write the **_exact_** output printed to standard output. Assume all variables are **int** variables with **x** and **y** initialized with values **2** and **4**, respectively.

```
1   z = x++*y;
2   printf("%d,%d,%d", x, y, z);
3
```

Answer: 3,4,8   ✔

Question 4

Correct

Points out of 1.00

Walk through the following code fragment and write the **_exact_** output printed to standard output. Assume all variables are **int** variables with **x** and **y** initialized with values **2** and **4**, respectively.

```
1  z = ++x*y;
2  printf("%d,%d,%d", x, y, z);
3
```

Answer: 3,4,12 ✔

Question 5

Correct

Points out of 1.00

Walk through the following code fragment to determine the value printed to the standard output stream and then pick the correct choice from the multiple choices. Assume all variables are of type **int** with **x** and **y** initialized to values **5** and **2**, respectively.

```
1  w = y*x + ++y;
2  printf("%d", w);
3
```

Select one:

○ 18

◉ the value printed to standard output stream is unspecified ✔

○ 13

○ 17

○ 14

Question 6

Correct

Points out of 1.00

Walk through the following code fragment to determine the value printed to the standard output stream and then pick the correct choice from the multiple choices. Assume all variables are of type **int** with **x** and **y** initialized to values **5** and **2**, respectively.

```
1  w = ++x * ++y;
2  printf("%d", w);
3
```

Select one:

○ 15

○ the value printed to standard output stream is unspecified

○ 10

○ 12

◉ 18 ✔

Question **7**

Incorrect

Points out of 1.00

Walk through the following code fragment to determine the value printed to the standard output stream and then pick the correct choice from the multiple choices. Assume all variables are of type **int** with **x** and **y** initialized to values **5** and **2**, respectively.

```
1  w = ++x + (x * ++y);
2  printf("%d", w);
3
```

Select one:

○ the value printed to standard output stream is unspecified

○ 18

○ 16

◉ 24 ✗

○ 20

Question **8**

Correct

Points out of 2.00

Walk through the following code fragment and write the **exact** output printed to standard output. Assume all variables are **int** variables with **i** and **j** initialized with values **54** and **23**, respectively.

```
1  k = i++ - 10 || --j;
2  printf("%d,%d,%d",++i,++j,++k);
3
```

Answer: 56,24,2 ✔

Question **9**

Correct

Points out of 5.00

Walk through the following code fragment and write the **exact** output printed to standard output. Assume variables **x** and **y** are of type **int** and initialized to values **100** and **200**, respectively.

```
1  if (x>100&&y<=200)
2      printf("%d",++x+y++);
3  else
4      printf("%d",2*x++- --y);
5
```

Answer: 1 ✔

Question **10**

Incorrect

Points out of 10.00

Consider the following statement:

```
1  b=2+++a;
2
```

There are two parts to this question:

- Determine the individual tokens in the above statement.
- Will the compiler consider this source line to be valid? If the source line is valid C code, write **y**; otherwise write **n**.

Write your answers (only) in this format: the comma-separated list of tokens followed by a comma followed by the character (**y** or **n**) indicating whether the source line is valid C code or not.

Answer: b,=,2,+,++,a,n ✗

Question 11

Incorrect

Points out of 10.00

Consider the following code fragment:

```
1  int x=5, y=6, z;
2  z=x+++ ++y; /* determine the number of tokens */
3  printf("%d", z);
4
```

There are three parts to this questions:

- Determine the tokens in the 2nd line of the code fragment.
- Will the compiler consider this source line to be valid? If the source line is valid C code, write **y**; otherwise write **n**.
- If you've determined that the 2nd source line is valid C code, then write the exact value printed to standard output.

Write your answers in this format: comma-separated list of tokens followed by a comma followed by a character (**y** or **n**) indicating whether the source line is valid C code or not. Only if you typed **y** then continue with a comma followed by the value printed to standard output.

Answer: z,=,x,++,+,++,y,y,12   ✗

Question 12

Correct

Points out of 1.00

Walk through the following code fragment and write the *exact* output printed to standard output:

```
1  int x=5, y=6+x++;
2  printf("%d", y);
3
```

Answer: 11   ✓

Question 13

Incorrect

Points out of 10.00

Consider the following code fragment where variables **x**, **y**, **z**, and **w** are **int** variables initialized with the values **5**, **3**, **0**, and **0**, respectively:

```
1  y = x++-++z;
2  z = 2*++y+3;
3  w = x++-2*++y+ ++z;
4  z = w++-++x;
5  w++;
6
```

What are the values of each of these variables after the last statement executes? Write the comma-separated variable values in this order: **x**, **y**, **z**, and **w**.

Answer: 8,5,-15,-5   ✗

Question 14

Correct

Points out of 10.00

Consider the following code fragment where variables **x**, **y**, and **z** are **int** variables initialized with the values **17**, **15** and **0**, respectively. Assume variables **t** and **w** are of type **double**.

```
1  z = x--+ --y/4;
2  y = x--%3+ ++z/4;
3  w = 17/3+6.5;
4  t = ++x/4.0+15%4-3.5;
5
```

Write the comma-separated variable values in this order: **x**, **y**, **z**, **t**, and **w**. Write the values of **double** variables rounded to **2** fractional digits. For example, **10.0** is written as **10.00** while **10.012** is written as **10.01**.

Answer: 16,6,21,3.50,11.50   ✓

Question **15**

Correct

Points out of
1.00

Walk through the following code fragment and write the *exact* output printed to standard output. Assume variables **x** and **y** are defined as type **int**.

```
1  x = 5;
2  y = 6 + ++x;
3  printf("%d", y);
4
```

Answer: 12   ✔

Question **16**

Correct

Points out of
5.00

Walk through the following code fragment and write the *exact* output printed to standard output. Assume variable **x** is defined as type **int** and is initialized to value **1**.

```
1  if (x>0)
2    switch (x) {
3      case 1:
4         x+=3;
5      case 3:
6         ++x;
7      case 6:
8         x++;
9      case 8:
10        x*=8;
11     default:
12        --x;
13    }
14 else
15    x+=2;
16 printf("%d", x);
17
```

Answer: 47   ✔

Question **17**

Correct

Points out of
2.00

Walk through the following code fragment and write the *exact* output printed to standard output. Assume **value** is defined as an **int** variable and initialized with the value **-1**.

```
1  switch(value) {
2    case -1: --value;
3    case 2: value++;
4    case 4: ++value;
5    case 5: value*=5;
6    case 6: value++;
7    default: --value;
8  }
9  printf("%d", value);
10
```

Answer: 0   ✔

Question **18**

Correct

Points out of
2.00

Walk through the following code fragment and write the **_exact_** output printed to standard output. Assume **value** is defined as an **int** variable and initialized with the value **3**.

```
1  switch(value) {
2      case 3: value+=3;
3      case 1: value++;
4      case 5: --value;
5      case 4: value+=4;
6  }
7  printf("%d", value);
8
```

Answer: 10   ✔

◄ Quiz 6: Iteration and Switch Statements   Jump to...   Lab 5: Flow Control ►