### Q1

1. 5‿‿,‿6.266
2. |1000,‿15|
3. NC: *format specifies 'int' but the argument is 'char \*'*
4. abcd,‿

### Q2

1. 'printf' returns the number of characters printed. If there are errors, 'printf' will return a negative value.

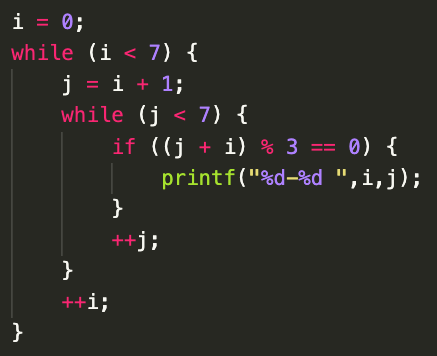
### Q3

1. x = 2, y = AMBI, z = 1
2. a = 6, b = AMBI
3. str = "hello", c = 32 (or '‿'), x = 6, y = 7.0f

### Q4

1. NC: *switch statement requires expression of integer type*
2. NC: *switch case must be an integer constant expression*
3. hello

### Q5

1. 
2. 0-3‿0-6‿1-2‿1-5‿2-4‿3-6‿4-5‿ (assuming i and j are defined)

### Q6

1. Is‿it‿me?:‿0

### Q7

1. 8,‿5,‿1

### Q8

1. 2‿6‿3‿8‿15‿

### Q9

1. 1:‿5,‿1,‿3,‿0,‿6
2. 1:‿1,‿1,‿2,‿1,‿3
3. 1:‿2,‿1,‿3,‿0,‿4

(note: can't compile if -Werror is used.

*'&&' within '||'* and *expression result unused*)

### Q10

1. (x > 6) ? (y += 7) : ((x < 3) ? (y -= 8) : 1);

(note: 1 can be substituted with anything)

y += (x > 6) ? 7 : ((x < 3) ? -8 : 0);

(alternative answer)

### Q11

1. (i < 3) && (i += 2);
2. (x <= y) && (x = y + 1);
3. (x < y) && (y = 5) || (y = x);

### Q12

1. int Foo (int a, float b);
2. void Boo (int \*a);
3. const float\* Goo (void);

(alternatively)

float const\* Goo (void);

1. void Doo (int \*const a);
2. void Hoo (double \*const \*a);
3. void Joo (float (\*a)[6]);
4. void Loo (float (\*const a)[]);

### Q13 (1)

1. int Foo[6] = { 0 };
2. int Goo[8] = { 1, 3, 5, 6 };
3. int Boo[100];

int i = 0;

for (i = 0; i < 100; ++i)

Boo[i] = 2;

1. int Goo[2][3] = {{ 1, 2, 0 }, { 3, 0, 0 }};
2. int Loo[100][10];

int i = 0;

for (i = 0; i < 1000; ++i)

Loo[0][i] = i;

### Q13 (2)

'ptr' is a pointer to an array of integers.

The size of int[] is unknown so you cannot perform (int[])\* arithmetic.

(i.e. the size of 'int[5]' = sizeof(int) \* 5 but size of 'int[]' = ???)

NC: *Arithmetic on a pointer to an incomplete type 'int []'.*

### Q14

1. int \*
2. int \*\*
3. int
4. int \*
5. 1. NC

2. int[3]

1. int \*
2. int
3. int(\*)[3]
4. int[2][3]
5. int \*
6. int(\*)[3]
7. int(\*)[2][3]

### Q15

1. -3
2. NC
3. 0
4. NC
5. 1006
6. NC
7. 1004
8. 1004
9. 998
10. 770 (or 0x0302)
11. 2
12. -9
13. NC

### Q16

*(note: int i is not an intended error)*

1. cs120isdead

### Q17

1. NC: *cannot increment value of type 'int [6]' (a.k.a. array)*

### Q18

1. Hello, Hellofalicious

### Q19

*(note: assuming stdlib is included)*

1. RTE: *Double Free and Dangling Pointer. Pointer is still pointing at an address that has been freed and tried to free that address again*
2. 8 bytes

### Q20

1. C
2. C
3. NC: *cannot assign to variable 'ipc' with const-qualified type 'int \*const'*
4. NC: cannot assign to variable 'ipc' with const-qualified type 'int \*const'
5. NC: *variable 'ip' is uninitialized when used here*
6. NC: *assigning to 'int \*' from 'const int \*' discards qualifiers*
7. C
8. C (NC with -Werror: *expression result unused*)
   1. NC: *read-only variable is not assignable*
9. C

### Q21

You cannot declare a struct of the same type within the struct

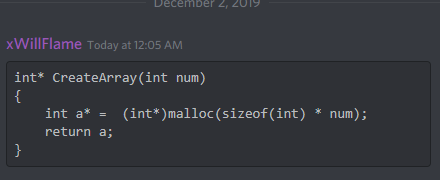
### Q22

(need TA to check)

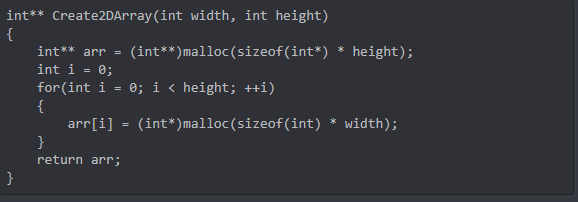
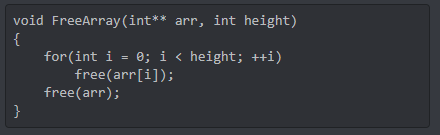
### Q23

1. 6
2. 32
3. 8
4. 1000 Check thru compiler
5. 1015
6. 1008
7. 1024
8. 1011
9. 1010

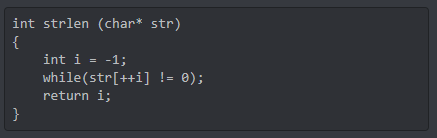
### Q24

1. 1. Should not return the address of the local variable. 2. Cannot declare a variable size array
2. 

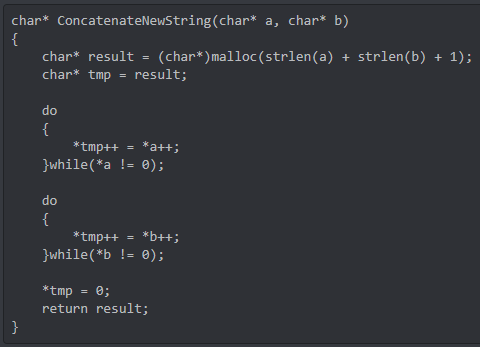
### Q25

1. 
2. 

### Q26



### Q27



### Q28

