

ECS 34 Final

Geoffrey Mohn

TOTAL POINTS

88 / 100

QUESTION 1

1 Q1 5 / 8

- 0 pts Correct
- 6.4 pts Blank
- ✓ - 3 pts Missed constructor
- 3 pts Missed operator*
- 2 pts Missed operator=

QUESTION 2

2 Q2 0 / 8

- 0 pts Correct
- 6.4 pts Blank
- ✓ - 4 pts i parameter should be pointer
- ✓ - 2 pts Didn't dereference i in foo
- ✓ - 2 pts Didn't take address of v in bar (or dynamically allocated v)

QUESTION 3

3 Q3 8 / 8

- ✓ - 0 pts Correct
- 6.4 pts Blank
- 4 pts %c is char
- 4 pts %d is int
- 1 pts List variables, not values.

QUESTION 4

4 Q4 8 / 8

- ✓ - 0 pts Correct
- 6.4 pts Blank
- 1 pts Close, issue is 'X' is char not a string as "X"
- 8 pts Incorrect, issue 'X' is char not null terminated string "X"
- 4 pts Don't need to allocate space

QUESTION 5

5 Q5 8 / 8

✓ - 0 pts Correct

- 6.4 pts Blank
- 1 pts 1 Wrong Edge
- 8 pts Multiple Wrong Edges
- 0.5 pts Wrong Total
- 3 pts No solid edges

QUESTION 6

6 Q6 8 / 8

- ✓ - 0 pts Correct
- 6.4 pts Blank
- 4 pts X = __FILE__, not X = __LINE__
- 8 pts Need to know file name

QUESTION 7

7 Q7 10 / 10

- ✓ - 0 pts Correct
- 8 pts Blank
- 0.5 pts Dijkstra's would be faster for no negative weights.
- 4 pts Use Bellman-Ford for negative weights
- 4 pts Use Dijkstra's multiple times if no negative weights

QUESTION 8

8 Q8 8 / 8

- ✓ - 0 pts Correct
- 6.4 pts Blank
- 2 pts need * sizeof(int)
- 8 pts Incorrect (should be int *Ptr = malloc(32 * sizeof(int));)
- 1 pts Don't need +1 like strings

QUESTION 9

9 Q9 8 / 8

- ✓ - 0 pts Correct

- **6.4 pts** Blank
- **8 pts** Incorrect, 7 bytes written, seek back 3 = 4
- **4 pts** Incorrect, there is space between foo bar and no null

QUESTION 10

10 Q10 9 / 10

- **0 pts** Correct
- **8 pts** Blank
- **2 pts** Reversed or missing arrows
- ✓ - **1 pts** Minor issue (.h not dep on .cpp, g++ not dep, etc.)
- **4 pts** Major issue (missing PROG, etc.)
- **10 pts** Incorrect

QUESTION 11

11 Q11 8 / 8

- ✓ - **0 pts** Correct
- **6.4 pts** Blank
- **4 pts** Incorrect order should be 22, 15, 6, 9, 10, 2, 1, 8

QUESTION 12

12 Q12 8 / 8

- ✓ - **0 pts** Correct
- **6.4 pts** Blank
- **4 pts** Not ideal, find MST
- **8 pts** Incorrect, should find MST


QUESTION 13

13 Notes 0 / 0

- ✓ - **0 pts** Correct
- **0.5 pts** Name wrong side
- **1 pts** Handwritten name.
- **2 pts** Wrong paper
- **2 pts** No notes

Final

I shall not cheat on, or knowingly give or receive assistance on this examination, and I shall not condone cheating by other persons. I shall not sit next to anyone (even if an empty seat is between us) that I am partnering with or have partnered with on any assignment, study with or have studied with regularly, or routinely spend time with socially. I understand that if I am suspected of academic misconduct that the Office of Student Judicial Affairs will handle the matter. I understand if I am found to be guilty of academic misconduct regarding this examination, or if I violate the rules on whom I sit next to, that I may receive a score of zero. If permitted to use handwritten notes, I agree to submit my notes when turning in this examination. I understand that if I leave a problem blank I will receive 20% of the credit for the problem.

Name (print): Geoffrey Andin Student ID: 912568148
Signature:  Date: 12/10/19
Student Left: End Student Right: Dorian Lin

Seat Number: G Left 5

Name: _____

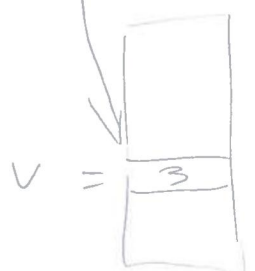
- 1) (8 points) Given the following C++ code, what functions have to be implemented by class C1?

```
int main(){
    C1 I, J{1}, K{2};
    I = J * K;
    return 0;
}
```

1. init
= Operator
** Operator*

- 2) (8 points) Translate the following C++ code into equivalent C code. Or Passes by reference?

```
void foo(int &i){
    i *= i;
}
void bar(void){
    int v = 3;
    foo(v);
}
```

Passed by ref
void foo(int &i) {
 *i *= i;*
}
void bar(void) {
 int v = 3;
 foo(v);
}
deferences?
code. Or Passes by reference?
foo()


- 3) (8 points) Fill in the blanks provided for the printf statement of which variables should be output.

```
char AChar = 'X';
int AnInt = 81;
double ADouble = 2.22;
```

%s *AChar = 'X'*
 printf("V1 = %d\nV2 = %c\n", AnInt, AChar);

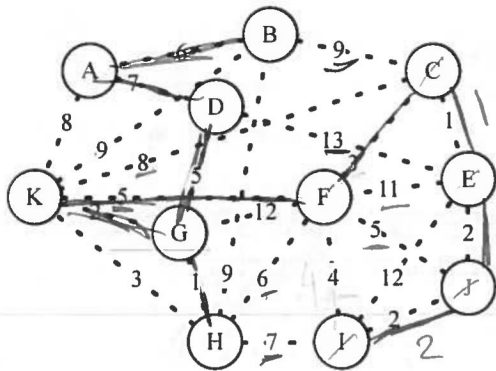
- 4) (8 points) A developer has written the following C code and it compiles but seg faults when it is run. What is wrong with the following code?

```
int main(){
    char *X = 'X';
    printf("X = %s\n", X);
    return 0;
}
```

**X Points to the address of 'X' with no null terminator in the memory block.*
*Need to allocate memory pointed for *X*

Name: _____

- 5) (8 points) Given the following graph, make the edges solid that are part of the minimum spanning tree. Fill out the total weight of the minimum spanning tree in the space provided.



I J E C F K G H D

B D A 10 15 20 27 28

2, 2, 1, 3, 5, 2, 1, 5, 7, 6 34

I J E C F K G H D A B

Total Weight: 34

- 6) (8 points) Given the following C++ code, put on the line what will be output? If it cannot be determined, leave output blank and then specify what information that is missing.

```

47: // Declare X
48: auto X = __FILE__;
49:
50: std::cout<<"X = "<<X<<std::endl;
51:

```

Output: _____ Missing: file name

- 7) (10 points) You are working on a project that needs to find all shortest paths from every source to every destination in a graph with no negative weights. How would you solve this problem most efficiently? Would your solution work if negative weights were allowed, but not negative cycles? If it will work, why will your solution work? If not, how would you solve the problem with negative edges?

I would use Dijkstra's algorithm but it would not work with negative edges, so I would implement Bellman Ford's algorithm for negative edges & no negative cycles.

- 8) (8 points) Given the following line of C++ code, give the equivalent line in C.

```
int *Ptr = new int[32];
```

```
int *Ptr = (int*) malloc(32 * sizeof(int));
```

Name: _____

- 9) (8 points) Assuming a char is one byte, what is the value of Offset after the following code is run?

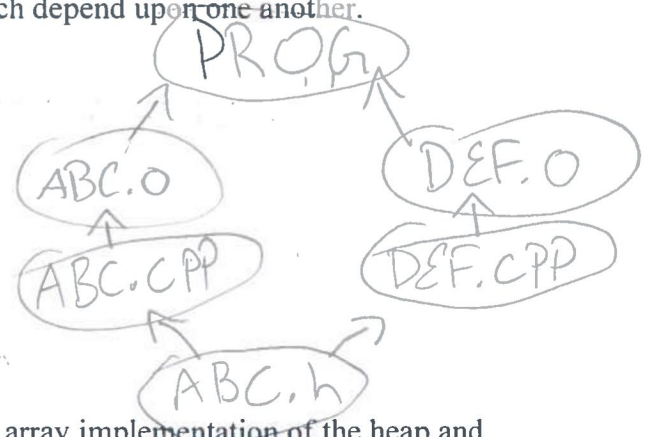
```
FILE *FP = fopen("file.txt", "w");
fwrite(FP, "foo bar", strlen("foo bar"));
fseek(FP, -3, SEEK_CUR);
Offset = ftell(FP);
Offset: 4
```

- 10) (10 points) Given the following snippet of a Makefile draw a Directed Acyclic Graph that represents all the files specified and which depend upon one another.

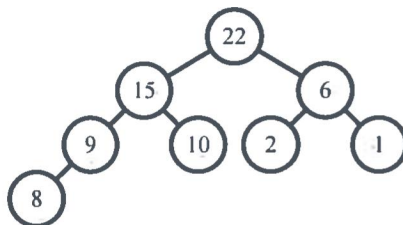
```
PROG: ABC.o DEF.o
g++ ABC.o DEF.o -o PROG
ABC.o: ABC.cpp ABC.h
g++ ABC.cpp -c -o ABC.o
DEF.o: DEF.cpp ABC.h
g++ DEF.cpp -c -o DEF.o
```

A → B
A depends on B
need B to make A
B → A

ABC.o ABC.cpp
DEF.o DEF.cpp
ABC.h



- 11) (8 points) Given the binary heap. Fill out the array implementation of the heap and circle if this is a min heap or max heap?



Index	1	2	3	4	5	6	7	8
Value	22	15	6	9	10	2	1	8

Min Heap

Max Heap

- 12) (8 points) Assume you were designing the original ARPANET (the predecessor to the Internet). You are tasked with connecting a set of cities together in the continental United States using as little money as possible (assume it costs \$10,000/mi to run a communication line). It has been decided that the communication lines will be run along existing highways in order to aid in maintainability. You are given a map of all cities involved, and the highways connecting them. How would you determine how the communication lines should be connected between the cities?

I would use Prim's Algorithm on a cyclic mst to get a min weighted connected graph that gets the shortest length of all visited nodes