**CSC175 Practice Assignment 7 Spring 2019 Name \_\_\_\_\_\_Donald Tvedt\_\_\_\_\_\_\_   
Directions:** Download this file and save as lastnamePracticeAssignment7SP19. Type all solutions on this document. Use equation editor when necessary. Upload Word document to Blackboard by **Friday at 11:59 AM.** Let me know if you have any questions. **Show Work!**

1) Use the following algorithms to sort the following list. Be sure to show and highlight each swap.

12 9 17 23 13 10  
a) Bubble sort

**12 9 17 23 13 10 – start sort**

**(9 12) 17 23 13 10 = sort 1**

**9 12 17 (13 23) 10 = sort 2**

**9 12 (13 17) 23 10 = sort 3**

**9 12 13 17 (10 23) = sort 4**

**9 12 13 (10 17) 23 = sort 5**

**9 12 (10 13) 17 23 = sort 6**

**9 (10 12) 13 17 23 = sort 7 Finished sorting**  
b) Selection sort

**12 9 17 23 13 10 – start sort**

**(9) (12) 17 23 13 10 = sort 1**

**9 (10) 17 23 13 (12) = sort 2**

**9 10 (12) 23 13 (17) = sort 3**

**9 10 12 (13) (23) 17 = sort 4**

**9 10 12 13 (17) (23) = sort 5 Finished sorting**

c) Insertion sort

**12 9 17 23 13 10 – start sort**

**(12) 9 17 23 13 10 = sort 1 start of the insertion sort 12 is selected**

**(9 12) 17 23 13 10 = sort 2**

**9 (12 17) 23 13 10 = sort 3 they are in sorted order**

**9 12 (17 23) 13 10 = sort 4 they are in sorted order**

**9 12 (13 17 23) 10 = sort 5 unsorted 13 moves to correct position in array bumping each over**

**9 (10 12 13 17 23) = sort 6 unsorted 10 moves to correct position in array bumping each over finished sorting**

d) Merge sort

**12 9 17 23 13 10 – start sort**

**(12 9 17) (23 13 10) = sort 1 split in half**

**(12 9) (17) (23 13) (10) = sort 2 split in half**

**(12) (9) (17) (23) (13) (10) = sort 3 split in half all singles move to sort portion**

**(9 12) (17 23) (10 13) = sort 4 merge into 2’s and sort each pair of 2**

**(9 12 17 23) (10 13) = sort 5 merge into 4’s and sort each pair of 4 remainders wait for next merge**

**(9 10 12 13 17 23) = sort 6 merge into 8’s and sort each pair of 8**

2) Use the algorithm to find the binary number for the decimal number 1453.

**1453 = 2\*726+1 List=1**

**726 = 2\*363+0 List =01**

**363 = 2\*181+1 list =101**

**181 = 2\*90+1 list =1101**

**90 = 2\*45+0 list = 01101**

**45 = 2\*22+1 list = 101101**

**22 = 2\*11+0 list = 0101101**

**11 = 2\*5+1 list = 10101101**

**5 = 2\*2+1 list = 110101101**

**2 = 2\*1+0 list = 0110101101**

**1 = 2\*0+1 answer= 10110101101**

3) Given the sets , find:  
a)   
b)   
c)   
d) Set builder notation for .  
e)

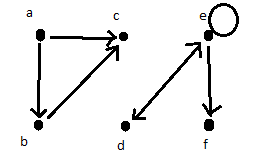
4) Use the binomial theorem to expand . Show all steps as shown on the class PowerPoint.

5) Create a truth table to show is a contradiction.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
| **T** | **T** | **F** | **T** | **F** | **F** |
| **T** | **F** | **T** | **T** | **F** | **F** |
| **F** | **T** | **F** | **F** | **T** | **F** |
| **F** | **F** | **T** | **T** | **F** | **F** |

6) Name the law used in each step.  
a) = **DeMorgan’s Law and involution**   
b) = **involution**  
c) = **Associative**  
d) = **Conjunctive Simplification**  
e) = **Conjunctive Simplification**  
Note: This is a proof for the truth table in #5. You will be writing these proofs yourself in MAT220. ☺

7) a) Use technology to create a digraph for the relation .

  
 b) Is S reflexive, symmetric, antisymmetric, and/or transitive? Why or why not?

**Not reflexive – all vertexes would need loopback, not symmetric – all arrows have to be double arrow, not antisymmetric – all arrows have to be single direction, not transtive – must create a loop between vertexes**

c) Create a table for the indegree and outdegree of each vertex.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **a** | **0** | **2** |
| **b** | **1** | **1** |
| **c** | **2** | **0** |
| **d** | **1** | **1** |
| **e** | **2** | **3** |
| **f** | **1** | **0** |

8) Use the Euclidean Algorithm to find GCD(810, 205) (show each step of the algorithm)

**810 = 3\*205+195**

**205 = 1\*195+10**

**195 = 19\*10+5**

**10 = 2\*5+0**

**GCD(810, 205) = 5**

9) Give the Big O runtime for the JustDoIt algorithm. Explain your answer.

Begin JustDoIt (List of numbers A)

Enter N (length of A)  
For i = 1 to N  
If A[i]<100 and A[i]>9  
 Print A[i]  
End If  
End For  
For j = 1 to N-1  
 For k=2 to N  
 If A[j]<A[k]  
 Print A[j]   
 End If  
 End For  
End For

End JustDoIt

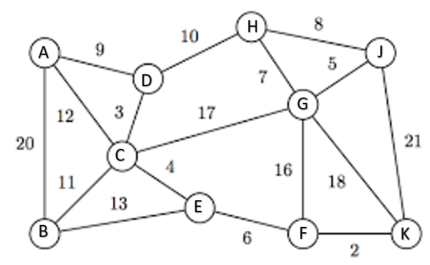
**Answer = , explanation is the nested for loops create the longest pull even though there is more to the equation its simplified to**

10) Explain what is happening in the JustDoIt algorithm in #9.

**In the first part of the code we are looping and printing every integer in the array that is greater then 9 and less then 100.**

**In the 2nd part of the code we are nested looping and comparing each consecutive number if the first number is less then the second number then print the first number. The first loop is traveling from index 1 to n-1 and the 2nd loop is traveling from index 2 to n.**

11) Use Prim’s algorithm to find the minimum spanning tree starting at vertex . List the edges in the correct order as found by the algorithm and give the sum.



**{{B,C}, {C,D}, {C,E}, {E,F}, {F,K}, {D,A}, {D,H}, {H,G}, {G,J}}**

**11 + 3 + 4 + 6 + 2 + 9 + 10 + 7 + 5 = 57**