Name _____

Multiple Choice. Write your answer to the LEFT of each problem. 3 points each

1. Suppose $f(x)$ is a monotonically increasing function. Which of the following approximates the summation?

A. $\int\limits_{m}^{n+1} f(x)dx \le \sum\limits_{k=m}^{n} f(k) \le \int\limits_{m-1}^{n} f(x)dx$ B. $\int\limits_{m-1}^{n-1} f(x)dx \le \sum\limits_{k=m}^{n} f(k) \le \int\limits_{m+1}^{n+1} f(x)dx$

C. $\int\limits_{m-1}^{n} f(x)dx \le \sum\limits_{k=m}^{n} f(k) \le \int\limits_{m}^{n+1} f(x)dx$ D. $\int\limits_{m}^{n} f(x)dx \le \sum\limits_{k=m}^{n} f(k) \le \int\limits_{m}^{n} f(x)dx$

2. Which of the following is solved heuristically by a greedy method?

    A. Fractional knapsack
    B. Finding the shortest paths from a designated source vertex in a sparse graph.
    C. Minimum spanning tree
    D. 0/1 knapsack

3. Which of the following dynamic programming problems maximizes its cost function?

    A. Optimal Matrix Multiplication
    B. Parking Permits
    C. Shuttle-to-Airport
    D. Weighted Interval Scheduling

4. What is the definition of H_n?

    A. $\Theta\left(\sqrt{n}\right)$ B. $\sum\limits_{k=1}^{n} k$ C. $\ln n$ D. $\sum\limits_{k=1}^{n} \frac{1}{k}$

5. Which of the following functions is not in $\Omega(n)$?

    A. $\lg n$ B. $n \lg n$ C. $n^2 \lg n$ D. n^3

6. Bottom-up heap construction is based on applying `fixDown` in the following fashion:

    A. In ascending slot number order, for each slot that is a parent.
    B. In descending slot number order, for each slot that is a parent.
    C. $\frac{n}{2}$ times, each time from the root of the heap.
    D. n - 1 times, each time from the root of the heap.

7. Suppose that a binary search is to be performed on a table with 100 elements. The maximum number of elements that could be examined (probes) is:

   A. 4   B. 5   C. 6   D. 7

8. Which of the following is a longest common subsequence for 0 1 2 0 1 2 and 0 0 1 2 1 2?

   A. 0 0 1 1 B. 0 1 2 1 2   C. 0 0 1 2   D. 0 1 2 0

9. The expected time for insertion sort for n keys is in which set? (All $n!$ input permutations are equally likely.)

   A. $\Theta(\log n)$ B. $\Theta(n)$ C. $\Theta(n \log n)$ D. $\Theta(n^2)$

10. What is the value of $\sum_{k=0}^{\infty} \left(\frac{1}{2}\right)^k$?

   A. $\frac{1}{2}$ B. $\frac{2}{3}$ C. $\frac{3}{2}$ D. 2

11. The purpose of the binary searches used when solving the longest (monotone) increasing subsequence (LIS) problem is:

   A. to assure that the final solution is free of duplicate values
   B. to determine the longest possible increasing subsequence terminated by a particular input value
   C. to search a table that will contain only the LIS elements at termination
   D. to sort the original input

12. Suppose you are using the substitution method to establish a Θ bound on a recurrence $T(n)$ and that you already know that $T(n) \in \Omega(\log n)$ and $T(n) \in O(n^2)$. Which of the following cannot be shown as an improvement?

   A. $T(n) \in O(\log n)$ B. $T(n) \in O(n)$ C. $T(n) \in \Omega(n^2)$ D. $T(n) \in \Omega(n^3)$

13. Which of the following is not true regarding a minheap with 1000 elements?

   A. Subscript 1 will store the maximum priority.
   B. The parent for the node with subscript 500 is stored at subscript 250.
   C. The left child for the node with subscript 200 is stored at subscript 400.
   D. The right child for the node with subscript 405 is stored at subscript 811.

14. The time to run the code below is in:

```
for (i=n; i>=0; i--)
  for (j=0; j<n; j+=2)
    sum+=i+j;
```

        A. $\Theta(n)$ B. $\Theta(n\log n)$ C. $\Theta\left(n^2\right)$ D. $\Theta\left(n^3\right)$

15. The time to run the code below is in:

```
sum=1;
for (i=1; i<n; i=3*i)
  sum++;
```

        A. $\Theta(\log n)$ B. $\Theta\left(\sqrt{n}\right)$ C. $\Theta(n)$ D. $\Theta\left(n^2\right)$

Short Answer:

1. Find a Huffman code tree for the following symbols and probabilities. The expected bits per symbol is NOT NEEDED. (5 points)

A	0.4
B	0.2
C	0.1
D	0.3

Long Answer

1. Complete the following example of the efficient dynamic programming technique for finding a Longest Increasing Subsequence (monotone). Circle the inputs that are in the final LIS. (10 points)

1	2	3	4	5	6	7	8	9	10	11	12
10	20	30	40	24	25	35	45	17	27	37	47

C	1	2	3	4								
j	0	1	2	3								

1	10/1

2	20/2

3   30/3

4   40/4

5

6

7

8

2. Use the substitution method to show that $T(n) = 3T\left(\frac{n}{3}\right) + n$ is in $\Theta(n\log n)$. 10 points

3. Use the recursion-tree method to show that $T(n) = 3T\left(\frac{n}{3}\right) + n$ is in $\Theta(n\log n)$. 10 points

4. Complete the following instance of the optimal matrix multiplication ordering problem, including the tree showing the optimal ordering. 10 points

p[0]=5   p[1]=4   p[2]=3   p[3]=4   p[4]=5   p[5]=6

		1		2		3		4		5
1	0	0	60	1	120	2	195	2	???	?
2	-------		0	0	48	2	120	2	222	2
3	-------		-------		0	0	60	3	150	4
4	-------		-------		-------		0	0	120	4
5	-------		-------		-------		-------		0	0

5. Show the maxheap after performing PQinsert for a priority of 22, followed by another PQinsert for a priority of 21. 10 points

CSE 2320-001   Name _____
Test 2
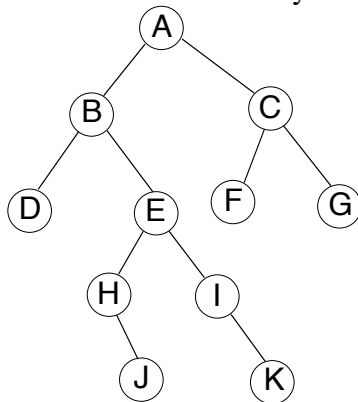Spring 2012   Last 4 Digits of Student ID # _____
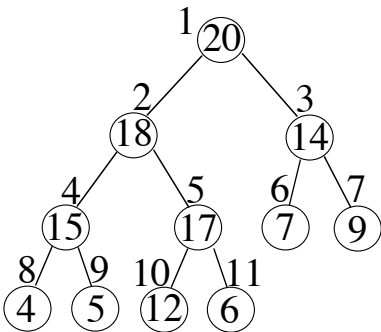
Multiple Choice. Write your answer to the LEFT of each problem. 3 points each

1. Which version of rat-in-a-maze finds the shortest path?

    A. recursive   B. red-black tree   C. stack   D. queue

2. Suppose the tree below is a binary search tree whose keys <u>are not shown</u>. Which node will contain the
    key that is the successor of the key stored at F?

A. A
B. B
C. C
D. G

3. Which of the following will not be true regarding the decision tree for HEAP-SORT for sorting n input
    values?

    A. Every path from the root to a leaf will have $O(n \log n)$ decisions.
    B. The height of the tree is $\Omega(n \log n)$.
    C. There will be $n!$ leaves.
    D. There will be a path from the root to a leaf with $\Omega\left(n^2\right)$ decisions.

4. What is the worst-case time to perform MAXIMUM(L) for a sorted, doubly-linked list with n nodes?

    A. $\Theta(1)$ B. $\Theta(\log n)$ C. $\Theta(n)$ D. $\Theta(n \log n)$

5. Which type of linked list will be the most convenient if LOGICALPREDECESSOR and
    LOGICALSUCCESSOR operations will be frequent?

    A. ordered, doubly-linked
    B. ordered, singly-linked
    C. unordered, doubly-linked
    D. unordered, singly-linked

6. Suppose a postfix evaluator has already processed 3 2 1 + * 4 5 + (with more to follow). What will be the contents of the stack (shown bottom-to-top going left-to-right)?

   A. 3 2 1 4 5
   B. 9 9
   C. 3 3 4 5
   D. 18

7. The most accurate description of the time to perform a deletion in an unbalanced binary search tree with n keys and height h is:

   A. $O(1)$ B. $O(\log n)$ C. $O(h)$ D. $O(n)$

8. Based on dictionary search performance alone, the best justification for ordering a linked list is:

   A. Many more misses than hits
   B. Sentinels are more effective
   C. Many more hits than misses
   D. Reduced storage

9. What is the worst-case time to perform PHYSICALPREDECESSOR(L, x) for an unsorted, singly-linked list with n nodes?

   A. $\Theta(1)$ B. $\Theta(\log n)$ C. $\Theta(n)$ D. $\Theta(n \log n)$

10. How should the successor of a node with a right child in an unbalanced binary search tree be found?

    A. Preorder traversal
    B. Examine the ancestors of the node
    C. Go left, then proceed to the right
    D. Go right, then proceed to the left

11. Which of the following binary trees can be legally colored as a red-black tree <u>with its root colored red</u>?

    A.   B.   C.   D.
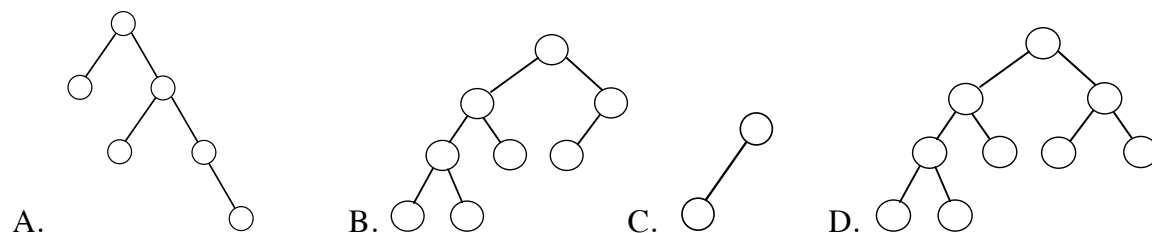
12. For which of the following sorts does the decision tree model not apply?

    A. Insertion   B. LSD Radix Sort   C. MERGE-SORT   D. QUICKSORT

13. What does counting sort count?

    A. the maximum length among all the strings being sorted
    B. the number of bytes in the input array
    C. the number of different input values that have occured
    D. the number of occurences for each possible key value

14. Which binary tree traversal corresponds to the following recursive code?

```
void traverse(node x)
{
if (x==null)
   return;
// process x here
traverse(x->left);
traverse(x->right);
}
```

    A. inorder   B. postorder   C. preorder   D. search for key x

15. Recently, we considered an abstraction supporting the operations *allocate*, *allocateAny*, and *freeup* which was implemented in constant time. Which of the following was not a feature of the implementation?

    A. a header
    B. a recycling list
    C. a circular, doubly-linked list
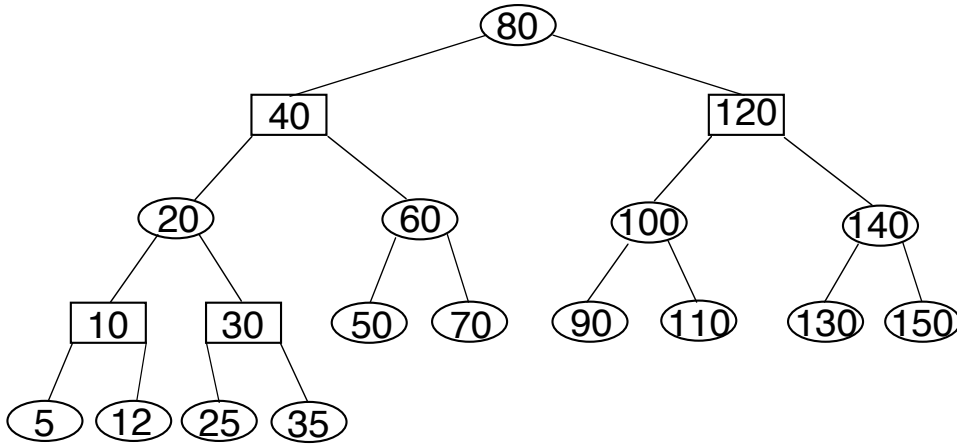   D. arrays

Long Answer

1. Give the unbalanced binary search tree that results when the keys 90, 50, 100, 80, 70, 60, 120 are inserted, in the given order, into an initially empty tree. (5 points)

2. Give code (or pseudocode) using PARTITION (iteratively) to place the k largest values in an unordered array of n values at subscripts $n - k \ldots n - 1$ of the same array. Your answer should take $\Theta(n)$ expected time. The k values may be unordered at the indicated subscripts. (10 points)

3. Show the result after PARTITION manipulates the following subarray. Be sure to circle which version of PARTITION you applied. (10 points)
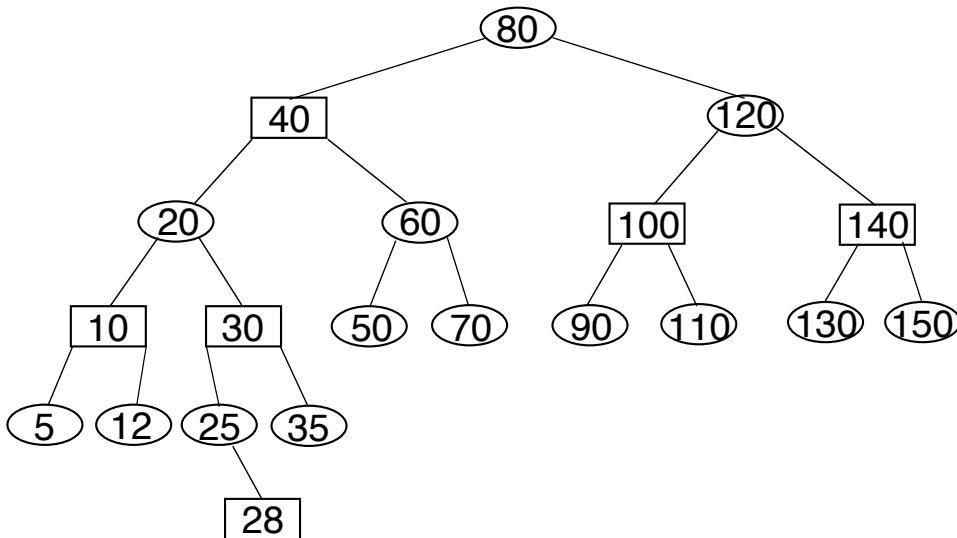
           8   2   5   3   7   4   1   9   0   6

   Version: 1   2/Sedgewick

4. Twenty (20) positive integers in the range 0 . . . 99,999,999 are to be sorted by LSD radix sort. Compare the performance for using radix 0 . . . 9999 and radix 0 . . . 9. Show your work. (10 points)

5. Insert 15 into the given red-black tree. Be sure to indicate the cases you used. (10 points)

```
                              80
                 40                        120
           20          60            100          140
        10    30    50   70        90   110    130   150
       5 12 25 35
```

6. Insert 26 into the given red-black tree. Be sure to indicate the cases you used. (10 points)

```
                              80
                 40                        120
           20          60            100          140
        10    30    50   70        90   110    130   150
       5 12 25 35
              28
```

CSE 2320-001   Name _____

Test 3

Spring 2012   Last 4 Digits of Student ID # _____

Multiple Choice. Write the letter of your answer to the LEFT of each problem. 2 points each

1. Suppose a depth-first search on a directed graph yields a path of tree edges from vertex X to vertex Y and a path of tree edges from vertex X to Z. If there is also an edge from Y to Z, then its type will be:

    A. Tree   B. Back   C. Cross   D. Forward

2. During a breadth-first search, the status of a black vertex is:

A. It has been completely processed.
B. It is in the FIFO queue.
C. It is in the priority queue.

    D. It is undiscovered.

3. Suppose a directed graph has a path from vertex X to vertex Y, but no path from vertex Y to vertex X. The relationship between the finish times is:

    A. finish(X) > finish(Y)   B. finish(X) < finish(Y)
   C. finish(X) = finish(Y)   D. could be either A. or B.

4. The capacity of any cut is:

    A. An upper bound on the maximum flow.   B. A lower bound on the maximum flow.
    C. The same as the capacity of all other cuts.   D. The same as the maximum attainable flow.

5. Suppose a maximum bipartite matching with k edges is found using Edmonds-Karp. Which of the following does not hold?

    A. The capacity of the minimum cut is k.   B. There will be $k + 1$ breadth-first searches.
    C. All residual network capacities are zero or one.   D. Every augmenting path uses three edges.

6. The capacity of the following cut is _____. (S vertices are bold.)
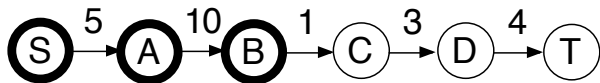
    A. 1   B. 10   C. 15   D. 23

7. Dijkstra's algorithm, when implemented with a heap, is most suitable for:

    A. Finding the minimum spanning tree of a dense graph.
    B. Finding the minimum spanning tree of a sparse graph.
    C. Finding the shortest paths from a designated source vertex in a dense graph.
    D. Finding the shortest paths from a designated source vertex in a sparse graph.

8. Before searching for a minimum cut in a network, it is useful to do the following:

    A. Find one augmenting path.
    B. Determine the type of each edge using depth-first search.
    C. Find and record augmenting paths until none remains.
    D. Perform a breadth-first search on the input network.

9. When finding the strongly connected components, the number of components is indicated by:

    A. The number of cross edges found during the second depth-first search.
    B. The number of back edges found during the first depth-first search.
    C. The number of restarts for the second depth-first search.
   D. The number of restarts for the first depth-first search.

10. When using two breadth-first searches to find the diameter of a tree, the purpose of the first search is to find:

A. the number of edges in the diameter.
B. all vertices that could be an end of a diameter.
C. both ends of a diameter.
D. one end of a diameter.

11. During depth-first search on an undirected graph, a cycle is indicated by which edge type?

   A. Back   B. Cross   C. Forward   D. Tree

12. What is the number of strongly connected components in this graph?
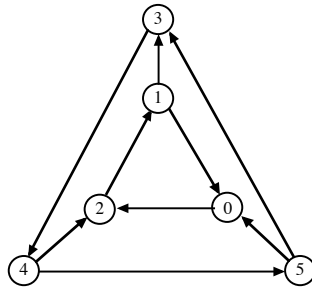
          A. 1   B. 2   C. 3   D. 4

13. Which edge is chosen in a phase of Kruskal's algorithm?

   A. The unprocessed edge (x, y) of smallest weight such that `find(x)==find(y)`
   B. An edge of maximum-weight in a cycle (to be excluded)
   C. An edge that is on a shortest path from the source
   D. The unprocessed edge (x, y) of smallest weight such that `find(x)!=find(y)`

14. Suppose a depth-first search is performed on an undirected graph with n vertices. The graph is a free (i.e. unrooted) tree if:

   A. all edges are tree edges
   B. both C and D
   C. there are no forward edges
   D. there are no cross edges

15. The number of potential probe sequences when using double hashing with a table with m entries (m is prime) is:

   A. $m-1$ B. m C. $m(m-1)$ D. $m!$

16. Suppose that there is only one path from vertex 5 to vertex 10 in a directed graph: $5 \rightarrow 7 \rightarrow 4 \rightarrow 3 \rightarrow 2 \rightarrow 10$. During the scan of which column will Warshall's algorithm record the presence of this path?

   A. 2   B. 3   C. 4   D. 7   E. 8   F. 10

Problems 17 and 18 refer to the following hash table whose keys are stored by double hashing using

$h_1(key) = key \% 13$ and $h_2(key) = 1 + (key \% 12)$.

0	1	2	3	4	5	6	7	8	9	10	11	12
			120	186	187	162		122	110			194

17. 266 would be inserted into which slot of the given table?

    A. 0   B. 1   C. 2   D. 7   E. 10   F. 11

18. 313 would be inserted into which slot of the given table?

    A. 0   B. 1   C. 2   D. 7   E. 10   F. 11

Problems 19 and 20 refer to the following hash table whose keys are stored by linear probing using $h(key) = key \% 13$.

0	1	2	3	4	5	6	7	8	9	10	11	12
			94		122	110	20	86	87	62		

19. 148 would be inserted into which slot of the given table?

   A. 0   B. 1   C. 2   D. 4   E. 11   F. 12

20. 133 would be inserted into which slot of the given table?

    A. 0   B. 1   C. 2   D. 4   E. 11   F. 12

Long Answer

1. What are the entries in the heap (for Prim's algorithm) before and after moving the next vertex and edge into the minimum spanning tree? DO NOT COMPLETE THE ENTIRE MST!!! Edges already in the MST are the thick ones. Edges currently not in the MST are the narrow ones. You do not need to show the binary tree for the heap ordering. 10 points.
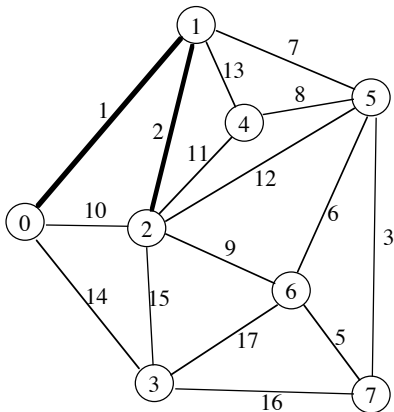

2. Perform depth-first search on the following graph, including start/finish times and edge types (T=tree, B=back, C=cross, F=forward.) Assume that the adjacency lists are ***ordered***. Write your answer in the tables below. 10 points
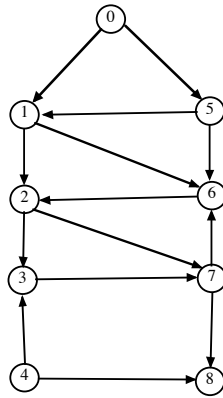
Vertex	Start	Finish		Edge	Type		Edge	Type
0	1	___		0 1	___		5 1	___
1	___	___		0 5	___		5 6	___
2	___	___		1 2	___		6 2	___
3	___	___		1 6	___		7 6	___
4	___	___		2 3	___		7 8	___
5	___	___		2 7	___			
6	___	___		3 7	___			
7	___	___		4 3	___			
8	___	___		4 8	___			

3. Show the *compressed* adjacency list representation this graph. (Answers using conventional adjacency lists will receive no credit.) 10 points.
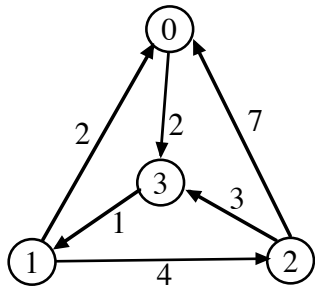
4. Give augmenting paths for determining a maximum flow and give a minimum cut for the following network. 0 is the source and 5 is the sink. 10 points.
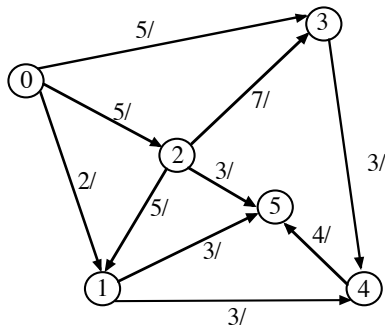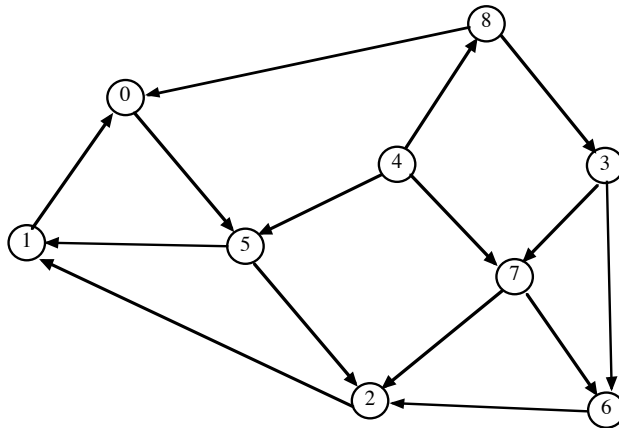
S vertices:

T vertices:

Augmenting Paths and Contribution to Flow:

5. Demonstrate, for the graph below, the algorithm that uses *two* depth-first searches to determine the strongly-connected components. 10 points

6. Give an algorithm (C code or pseudocode), based on the Floyd-Warshall algorithm with *successors*, to determine for "all pairs" of starting and destination vertices the path that *minimizes* the sum of the intermediate vertex numbers. So, if there is an input edge from vertex i to vertex j, then the sum would be zero. If there were a path $10 \rightarrow 20 \rightarrow 15 \rightarrow 12 \rightarrow 3$, its sum would be $20 + 15 + 12 = 47$. As usual, vertex numbers start at 0, so the use of vertex 0 as an intermediate vertex is "free". (10 points)