1. In the following example, what is the value of the max-weight independent set, and that of the output of our greedy algorithm

1-4-5-4

1. 14 and 10
2. 8 and 6
3. 8 and 8
4. 9 and 8

(2) is the correct answer. The maximum indepedent set are the vertices with weights 4 and 4. The greedy algorithm would select the largest weight vertex and from there select any maximum weight vertex that is still valid. This would result in a total weight of 6. This shows that a greedy approach the problem will not be optimal

2. How many different independent sets does a complete graph with 5 vertices have? How about a cycle with 5 vertices?

1. 1 and 2
2. 5 and 10
3. 6 and 11
4. 6 and 16

(3) is the correct answer. For the complete graph once we have chosen a vertex we have no other vertex that can be joined alongside it. Since we have 5 choices the amount of independent sets with at least one vertex is 5. We now include the independent set with no vertices for a total of 6 independent sets for a graph complete graph with 5 vertices. For a cycle graph with 5 vertices we can again use the same analysis to get 6 independent sets but now we have the options of getting independent sets with at least two members. The total number of independent sets with at least two elements is 5 hence we get a total of 11 independent sets for a cycle graph with 5 vertices.

3. What is the total weight of the output of the greedy algorithm when the input graph is the four-vertex path? Is this the maximum possible?

1. 6;no
2. 6;yes
3. 8;no

4. 8;yes

(1) is the correct answer. We the algorithm would first choose the 5 and then choose the 1 to get a total of 6. But the maximum independent set has a total weight of 8 implying that the greedy algorithm fails to solve the problem correctly.

4. What is the asymptotic running time of the recursive *WIS* algorithm, as a function of the number n of vertices?

1. $O(n)$
2. $O(n \cdot logn)$
3. $O(n^2)$
4. none of the above

(4) We have a recurrence of $T(n) = T(n-1) + T(n-2) + \Theta(1) \geq F_n$ where F_n is nth fibonacci number. This simplifies to $T(n) \geq \Theta(\phi^n)$ where ϕ is the golden ratio.

5. Each of the recursive calls of the recursive WIS algorithm is responsible for computing an MWIS of a specified input graph. Ranging over all of the calls, how many *distinct* input graphs are ever considered?

1. $\Theta(1)$
2. $\Theta(n)$
3. $\Theta(n^2)$
4. $2^{\Theta(n)}$

(2) is the correct answer we compute the values for $G_n, G_{n-1} \ldots, G_0$ which in total is $n+1$ graphs. This is order of n and hence is $\Theta(n)$.