Quiz 1

1. An employer offers a job to a student it has never made an offer to before.

2. Once a student has a job, they will remain hired for the remainder of the execution of the algorithm.

   If a student is hired by the first employer on their preference list at some point in the execution of the algorithm, then the last employer on their preference list has not yet made them an offer.

   It is possible for every employer and student to be paired with their top choice.

3. 10^5.

4. $O(2^n + n) = O(2^n)$.

   $O(3n + 100) = O(n)$.

   $O(n^2 + n) = O(n^2)$.

5. Inserting an element into a descending array of size n and building a new descending array.

   Merging two ascending arrays of size n to build a new ascending array.

6. $\Theta(n^2)$.

Quiz 2

1. $O(n \log n)$.

2. No, this approach is not correct: the value k is not a constant in this proof and changes each time.

3. $O(n)$, $O(n^2)$.

4. Yes. When merging, we compare the smallest value of first half of elements with the smallest value of the second half of elements and, add the minimum of these two to the output and erase if from its list. This takes time $O(n)$.

Quiz 3

1. This strip might contain in $\Omega(n)$ points.

2. Yes. We only need to calculate the distance of each point with a small subset of other points. This takes time $O(n)$.

Quiz 4

1. $2^{n/2}$
2. S1 is always valid, but S2 can sometimes be invalid
3. Sometimes, but not always
4. The set of the starting times of all the intervals

   The set of the finishing times of all the intervals
5. 8
6. Lateness can either increase, decrease, or stay the same:

   Decrease : Imagine that job A is currently scheduled before job B, but job B's deadline is earlier than job A's. Then, after we swap them, job B might not even be late, and job A will be less late than job B was before.

   Same : Imagine A and B had the same deadlines, then whichever is now the later job has the same lateness as whichever used to be the later job.

   Increase : Imagine that job A is currently scheduled before job B, and job B's deadline is later than job A's. Then, in the new order, job A will end at the same time that job B used to end, and job A will be further past its deadline than: (1) job A was before the swap, (2) job B was before the swap, and (3) job B is after the swap. So, lateness could increase!
7. Schedule S' may not be an optimal schedule.

   Some of the steps of the construction of S' may result in a non-reduced schedule.

   If S and S_{FF} evict items f and e respectively, with $e \neq f$, in step $j + 1$, then the schedule S' will have the same cache content as S by the time there is a request for e.
8. Bring in fewer or the same number of items as S.

Quiz 5

1. A = 001, B = 010, C = 011, D = 100, E = 101, F = 110

   A = 00, B = 01, C = 10, D = 110, E = 11100, F = 11111

   A = 0, B = 100, C = 101, D = 110, E = 1110, F = 1111
2. A deaf bee faced a faded facade
3. If a letter x is closer to the root of the tree than a letter y, then x's frequency is either higher or the same as y's frequency

   The tree is constructed from the leaves up.

Quiz 6

1. GreedyEarliestFinish: 8

   GreedyHighestValue: 7

   Optimal: 9

2. For any j between 1 and n, if we know the optimal scheduling for the first $j-1$ intervals, finding the optimal solution for the first j intervals amounts to making a binary choice.

   The optimal schedule that includes the nth interval adds its value to the optimal solution for the first $p(n)$ intervals.

   The original problem instance can be expressed as a subproblem of itself

3. 1

   6

4. By the time we consider a particular problem, whether the solutions to its subproblems have already been computed.

Quiz 7

1. There are a linear number of subproblems to solve, each of which takes linear time to solve.

2. $\mathrm{OPT}(i, w) \geq \mathrm{OPT}(i-3, w)$

   $\mathrm{OPT}(i, w) \geq \mathrm{OPT}(i, w-a)$

   $\mathrm{OPT}(i, w) \geq \mathrm{OPT}(i-1, w-w_i)$

3. 10

Quiz 8

1. Minimum alignment cost of X[1...i] and Y[1...j]

2. $F(i-1, j)$, $F(i, j-1)$ and $F(i-1, j-1)$.

Quiz 9

1. "Problem Y is polynomial-time reducible to problem X" is equivalent to "problem X is at least as hard as problem Y with respect to polynomial time".

Assume problem Y is polynomial-time reducible to problem X. If X can be solved in polynomial time, Y can be solved in polynomial time.

Let $G = (V, E)$ be a graph. S is an independent set if and only if $V - S$ is a vertex cover.

2. A list of truth assignments to each variable in the problem.

3. If a problem belongs to $\mathcal{NP}$ then every Yes instance of this problem admits an efficient verifier.

   Every problem in the class $\mathcal{NP}$ can be solved in time $O(2^{f(n)})$ where f is a polynomial function of the instance size n.

4. Reduce a known NP-complete problem B to A and show that A is in NP

Quiz 10

1. Directions along the paths that represent a node correspond to the two possible truth values for that variable.

   If there is a Hamiltonian Circuit in the graph constructed by the reduction, then it visits all of the nodes representing variable X_i before visiting the nodes representing variable X_{i+1}.

   The nodes of the graph that represent the clauses of the 3SAT instance have degree 6.

2. The gadget associated with variable X_i in the reduction contains 4 times as many elements as there are clauses in the instance of 3SAT.

   The reduction from 3SAT to 3DM adds three triples for each clause in the instance of 3SAT.

   The total number of triples in the reduction is $2kn + 3k + (n-1)k$, where n is the number of variables and k is the number of clauses in the instance of 3SAT.

Quiz 11

1. Independent Set

   Set Packing

   Vertex Cover

   Traveling Salesperson

2. The problem can be solved in polynomial time for some values of k.

   Graph coloring has applications in compiler design.

Every variable in the 3SAT instance is represented by two vertices in the graph constructed by the reduction.

3. The purpose of the vertex labeled B is to ensure that only two colors are used for the vertices labeled with variable names.

   Every variable in the 3SAT instance is represented by two vertices in the graph constructed by the reduction.

4. There are scheduling problems that can be proved NP-complete using a reduction from Subset Sum.

5. Numbers in the instance of Subset Sum generated by the reduction are written in base $m + 1$, where m is the number of triplets in the instance of 3DM.

   Each triplet in the instance of 3DM is represented by a $3n$ digits integer in the instance of Subset Sum generated by the reduction, where n is the size of the sets if the instance of 3DM.

   The construction of the instance of Subset Sum ensures the addition will not produce carries.

Quiz 12

1. Getting a good worst-case bound on a sequence of operations.

2. $\Phi(D_i) \geq 0$

   For any given sequence of operations on a data structure, there may be many potential functions LaTeX: $\Phi(D_i)$ that will allow us to prove an optimal bound on these operations.

3. We compute an upper bound on the sum of the amortized costs of the n operations in order to get an upper bound on the sum of their real costs.

   The amortized cost of an operation can be smaller, the same or larger than its real cost.

   Ideally, the amortized cost of an operation should vary relatively little.

4. Both the real cost and the amortized cost are constants.

   The amortized cost is $\log_2 n$ while the real cost varies between 1 and n.