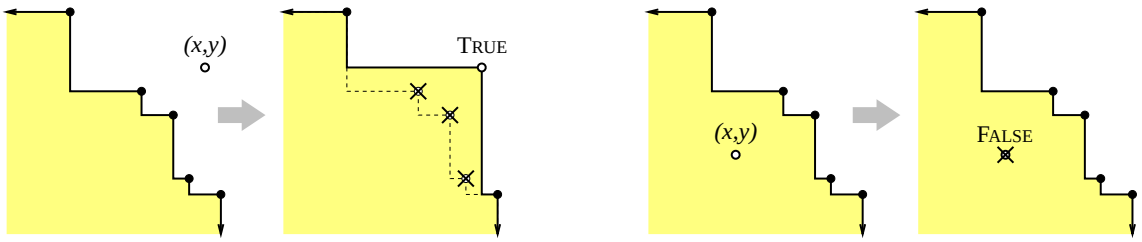
**You must turn in this question sheet with your answers.**

1. **Déjà vu**

   Prove that any positive integer can be written as the sum of distinct *nonconsecutive* Fibonacci numbers—if $F_n$ appears in the sum, then neither $F_{n+1}$ nor $F_{n-1}$ will. For example: $42 = F_9 + F_6$, $25 = F_8 + F_4 + F_2$, and $17 = F_7 + F_4 + F_2$. *You must give a complete, self-contained proof, not just a reference to the posted homework solutions.*

2. **L'esprit d'escalier**

   Recall that the *staircase* of a set of points consists of the points with no other point both above and to the right. Describe a method to maintain the staircase as new points are added to the set. Specifically, describe and analyze a data structure that stores the staircase of a set of points, and an algorithm INSERT$(x, y)$ that adds the point $(x, y)$ to the set and returns TRUE or FALSE to indicate whether the staircase has changed. Your data structure should use $O(n)$ space, and your INSERT algorithm should run in $O(\log n)$ amortized time.




3. **Engage le jeu que je le gagne**

   A palindrome is a text string that is exactly the same as its reversal, such as `DEED`, `RACECAR`, or `SAIPPUAKAUPPIAS`.[1]

   (a) Describe and analyze an algorithm to find the longest *prefix* of a given string that is also a palindrome. For example, the longest palindrome prefix of `ILLINOISURBANACHAMPAIGN` is `ILLI`, and the longest palindrome prefix of `HYAKUGOJYUUICHI`[2] is the single letter `S`. For full credit, your algorithm should run in $O(n)$ time.

   (b) Describe and analyze an algorithm to find the length of the longest *subsequence* of a given string that is also a palindrome. For example, the longest palindrome subsequence of `ILLINOISURBANACHAMPAIGN` is `NIAACAAIN` (or `NIAAHAAIN`), and the longest palindrome subsequence of `HYAKUGOJYUUICHI` is `HUUUH`[3] (or `HUGUH` or `HUYUH` or...). You do not need to compute the actual subsequence; just its length. For full credit, your algorithm should run in $O(n^2)$ time.

[1] Finnish for 'soap dealer'.
[2] Japanese for 'one hundred fifty-one'.
[3] English for 'What the heck are you talking about?'

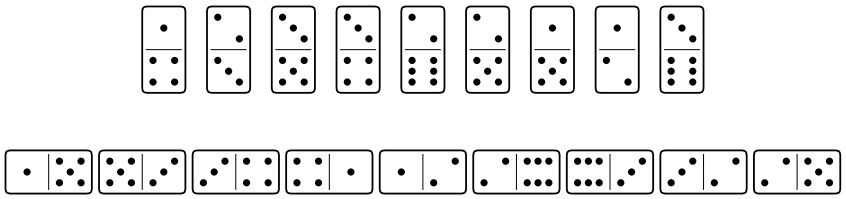
4. **Toute votre base sont appartiennent à nous**

   Prove that *exactly* $2n - 1$ comparisons are required in the worst case to merge two sorted arrays, each with $n$ distinct elements. Describe and analyze an algorithm to prove the upper bound, and use an adversary argument to prove the lower bound. *You must give a complete, self-contained solution, not just a reference to the posted homework solutions.*[4]

5. **Plus ça change, plus ça même chose**

   A domino is a $2 \times 1$ rectangle divided into two squares, with a certain number of pips (dots) in each square. In most domino games, the players lay down dominos at either end of a single chain. Adjacent dominos in the chain must have matching numbers. (See the figure below.)

   Describe and analyze an efficient algorithm, or prove that it is NP-hard, to determine whether a given set of $n$ dominos can be lined up in a single chain. For example, for the set of dominos shown below, the correct output is TRUE.



Top: A set of nine dominos
Bottom: The entire set lined up in a single chain

6. **Ceci n'est pas une pipe**

   Consider the following pair of problems:

   - BOXDEPTH: Given a set of $n$ axis-aligned rectangles in the plane and an integer $k$, decide whether any point in the plane is covered by $k$ or more rectangles.
   - MAXCLIQUE: Given a graph with $n$ vertices and an integer $k$, decide whether the graph contains a clique with $k$ or more vertices.

   (a) Describe and analyze a reduction of one of these problems to the other.

   (b) MAXCLIQUE is NP-hard. What does your answer to part (a) imply about the complexity of BOXDEPTH?

7. **C'est magique!** *[1-unit graduate students must answer this question.]*

   The recursion fairy's cousin, the reduction genie, shows up one day with a magical gift for you—a box that determines in constant time the size of the largest clique in any given graph. (Recall that a clique is a subgraph where every pair of vertices is joined by an edge.) The magic box does not tell you *where* the largest clique is, only its size. Describe and analyze an algorithm to actually find the largest clique in a given graph **in polynomial time**, using this magic box.

[4]The posted solution for this Homework 5 practice problem was incorrect. So don't use it!