# CIS 1210 — Data Structures and Algorithms
# Homework Assignment 9

**Assigned:** November 19, 2024 **Due:** November 27, 2024

---

**Note:** The homework is due **electronically on Gradescope** on November 27, 2024 by 11:59 pm ET. For late submissions, please refer to the Late Submission Policy on the course webpage. You may submit this assignment up to 2 days late.

**A. Gradescope**: You must select the appropriate pages on Gradescope. Gradescope makes this easy for you: before you submit, it asks you to associate pages with the homework questions. Forgetting to do so will incur a 5% penalty, which cannot be argued against after the fact.

**B. LaTeX**: You must use the LaTex template provided on the course website, or a 5% penalty will be incurred. Handwritten solutions or solutions not typeset in LaTex will not be accepted.

**C. Solutions**: Please write concise and clear solutions; you will get only a partial credit for correct solutions that are either unnecessarily long or not clear. Please refer to the Written Homework Guidelines for all the requirements. Piazza will also contain a complete sample solution in a pinned post.

**D. Algorithms**: Whenever you present an algorithm, your answer must include 3 separate sections. Please see Piazza for an example complete solution.

1. A precise description of your algorithm in English. No pseudocode, no code.
2. Proof of correctness of your algorithm
3. Analysis of the running time complexity of your algorithm

**E. Collaboration**: You are allowed to discuss **ideas** for solving homework problems in groups of up to 3 people but *you must write your solutions independently.* Also, you must write on your homework the names of the people with whom you discussed. For more on the collaboration policy, please see the course webpage.

**F. Outside Resources**: Finally, you are not allowed to use *any* material outside of the class notes and the textbook. Any violation of this policy may seriously affect your grade in the class. If you're unsure if something violates our policy, please ask.

**1. [20 pts] Dhruv's Volleyball Stash Table**

After the 1210 TA IM Volleyball team lost their last round robin game in a tiebreaker that went to deuce, Dhruv Goyal had been practicing his sets, spikes, and serves nonstop in the gym. However, he had been short on equipment. So, when Black Friday came around, Dhruv was **fiending** to purchase volleyballs, knee pads, setting machines, and more.

After coming back from the Dick's Sporting Goods with his stuff, he lines up $m = 14$ boxes to stash the items. He decides to treat the line of boxes as a hash table, hashing each item by its price.

In order to access the items quickly (so he can get to playing volleyball as quickly as possible), he wants to try a few different hashing strategies. Given the prices 8, 54, 12, 32, 89, 180, 43, and 5 as keys (in that order), insert them into the stash table of length $m = 14$ using open addressing with the auxiliary hash function $h'(k) = k$. Give the resulting hash table after inserting these keys using:

**a.** Linear probing

**b.** Quadratic probing with $c_1 = 2$ and $c_2 = 5$

**c.** Double hashing with $h_1(k) = k$ and $h_2(k) = 7 + (k \bmod (m - 3))$

**2. [20 pts] Kevin's Jellycat Spree**

To fulfill his new obsession for Jellycat stuffed animals, Kevin Zhou has developed a plan to buy as many cute plushies as possible at a sweet discount on Black Friday. In an effort to maximize his efficiency, Kevin has recruited $n$ friends, where $n$ is an even positive integer. In a frenzied rush each friend starts grabbing at least one plushy for a total of $kx$ plushies, where $x$ is a positive integer. However, as Kevin's friends pile into the checkout line they find that corporate America has unsurprisingly added a hidden catch behind the discount.

More specifically, the store requires all plushies to be placed into a basket, and the Black Friday discount only applies to baskets containing a multiple of $k$ plushies. Additionally, exactly two people can checkout together with one basket. In order for Kevin's operation to be both bountiful and discounted, he must find a way to assign his friends into $\frac{n}{2}$ pairs such that the sum of their plushies in their basket is a multiple of $k$.

Design an algorithm that runs in **expected** $O(n)$ time to determine if it is possible for Kevin to pair up his friends such that each friend is assigned to exactly one pair, and **all pairs' baskets** have a multiple of $k$ plushies.

For example, if the checkout constraint requires each basket to be divisible by $k = 10$, and there are a total of $kx = 150$ plushies, and the group distribution is as follows:

| Friend: | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| Number of Plushies: | 30 | 12 | 10 | 26 | 18 | 54 |

Here, Kevin should pair up friends $(A, C)$ which will total to 40 plushies in their basket, groups $(B, E)$ should pair up which will total 30 plushies in their basket, and groups $(D, F)$ should pair up to total 80 plushies in their basket. Thus, since the count of plushies in each basket is divisible by $k = 10$, your algorithm should return true. If we adjust the assignment such that friend $F$ has 56 plushies and friend $E$ has 16 plushies, then your algorithm should return false.

**3. [20 pts] Eloic's Gift Shopping Puzzle**

It's past October 31st, so far as Eloic is concerned, it's Christmas time! For Black Friday, he has a list of gifts he wants to prepare for his fellow TAs. However the mall at which he plans to shop has decided to make it a challenge by hiding their catalogue in an $n$ by $n$ word search puzzle with characters ranging from A-Z. The word bank, which contains the names of the gifts he wants to find, has at most $n^2$ total words, where each word has at most $k$ characters (you may assume $k$ is some non-constant variable $\leq n$). Worried that he won't find everything, Eloic wants to determine if all of the gifts on his list will be at the mall. Note that a gift is considered to be in the word search puzzle if each of its characters appear in order and in a straight line (words can appear horizontally, vertically, or diagonally, and in both forward and backward direction).

In order to beat the crowd, Eloic needs to figure things out fast. Help him come up with a $O(n^2k)$ algorithm that returns true if all gifts are in the catalog, and false otherwise!

For example, given the word bank [BLANKET, CARD, BOOK, TROLLEY, ROOMBA] and the following word search puzzle, the algorithm should return that Eloic will find everything:

| | | | | | | |
|---|---|---|---|---|---|---|
| B | K | B | T | A | O | T |
| L | I | K | A | B | N | R |
| A | G | T | O | M | D | O |
| N | O | R | T | O | P | L |
| K | B | O | E | O | B | L |
| E | X | C | A | R | D | E |
| T | K | C | A | X | Q | Y |