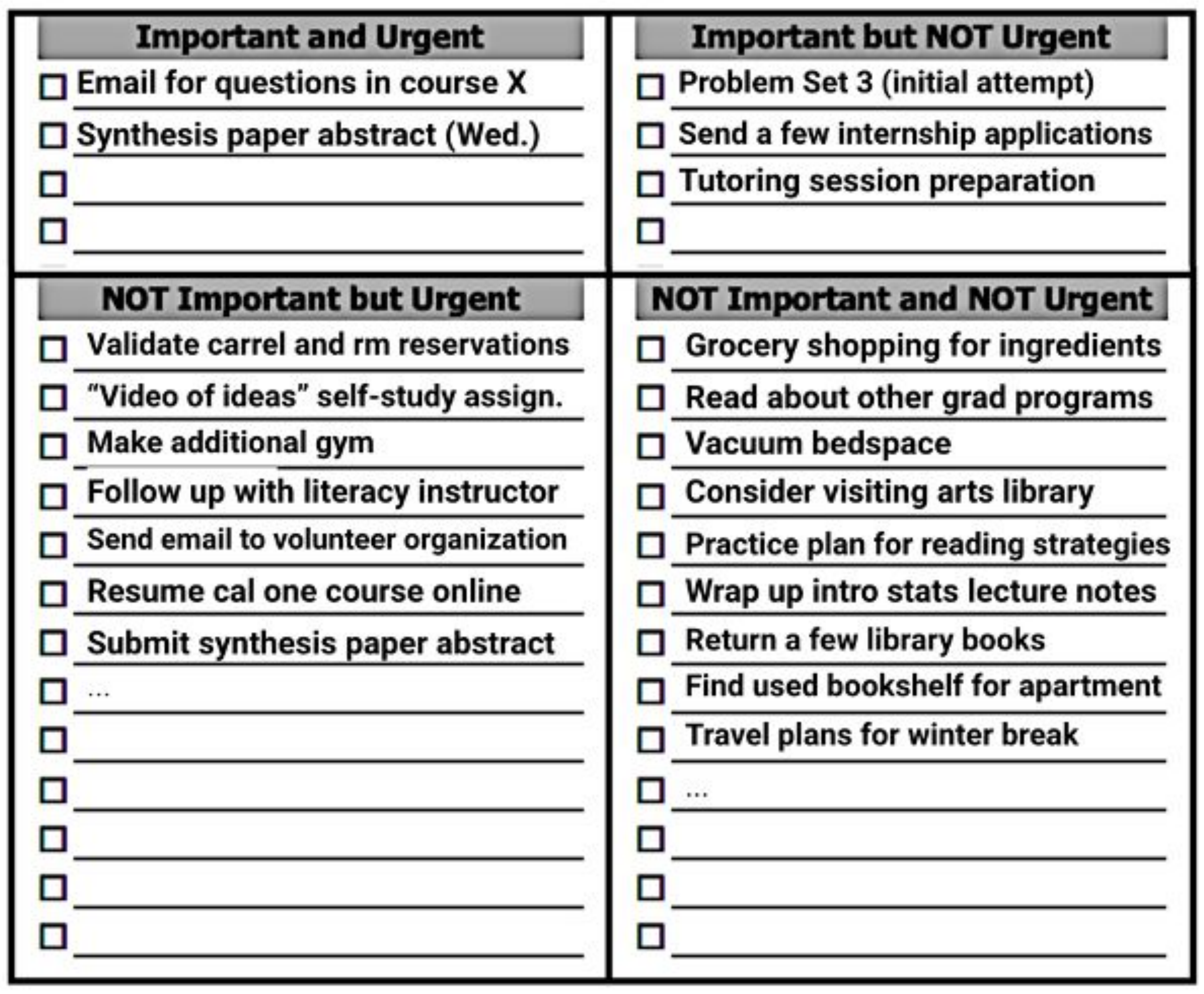
A Brief Survey of Elementary Algorithmic Literacy

Dachao Sun University of Pennsylvania Philadelphia, Pennsylvania

Introduction

The ability to **dismantle a problem** and **think algorithmically** has shown to be more and more relevant and effective in modern life. The basic capability to leverage algorithms, or what we refer to as "algorithmic literacy"—beyond the reformation of the educational landscape based on computer usage—is useful to many fields and requires two general components or stages of development.

- Training of basic algorithmic techniques on **abstract mathematical structures**, as instructed in a common modern course of high school or collegiate computer science curriculum;
- Translation for the practice of problem solving skills in social settings.



Important and Urgent

□ Email for questions in course X
□ Synthesis paper abstract (Wed.)
□
□

Important but NOT Urgent

□ Problem Set 3 (initial attempt)
□ Send a few internship applications
□ Tutoring session preparation
□

NOT Important but Urgent

□ Validate carrel and rm reservations
□ "Video of ideas" self-study assign.
□ Make additional gym
□ Follow up with literacy instructor
□ Send email to volunteer organization
□ Resume cal one course online
□ Submit synthesis paper abstract
□ ...
□
□
□
□

NOT Important and NOT Urgent

□ Grocery shopping for ingredients
□ Read about other grad programs
□ Vacuum bedspace
□ Consider visiting arts library
□ Practice plan for reading strategies
□ Wrap up intro stats lecture notes
□ Return a few library books
□ Find used bookshelf for apartment
□ Travel plans for winter break
□ ...
□
□
□

Algorithmic Literacy

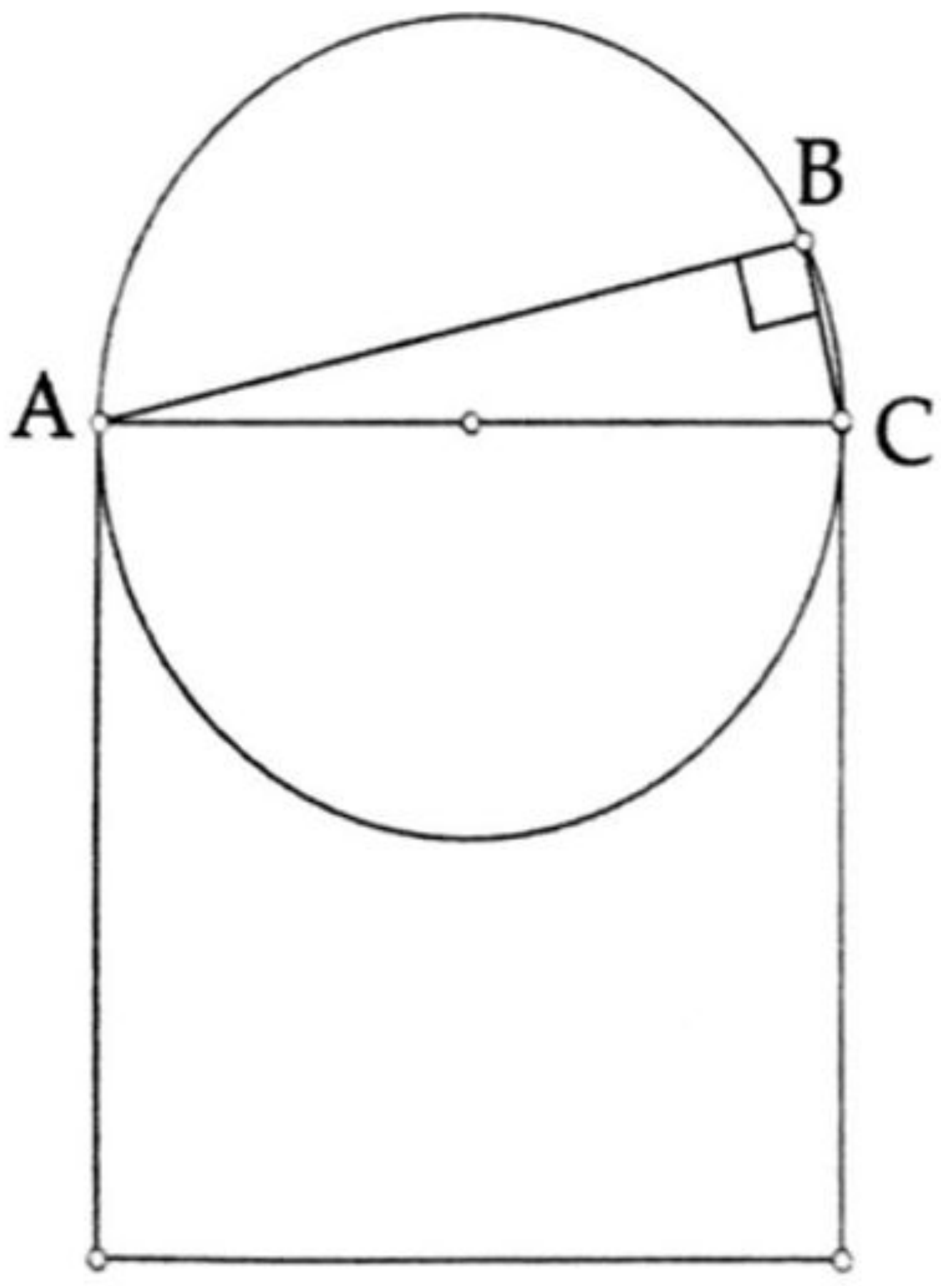
Algorithmic literacy can serve as one of the bridges to the science of decision for the pursuit of "skillful judgment formation and decision-making" (Alliance for Decision Education, 2022).

The Concept of Data / Fundamental Data Structures

- Data is pervasive and can be anything: several photographs of a cat could be considered a "dataset".
- In most practical situations, though, we focus on and leverage **numeric** data, data that consist of certain types of numbers primarily integers and real numbers.
- An important feature of such data that is worth noting is that they are *numerable*, or that they can be **compared** and **traversed** or "enumerated".
- A typical, abstract type of data structures is called a set or a **dictionary**, one that supports the following operations,

1. Insert: to add a new element into some/any place within the data structure;
2. Remove: to take away a specific element from data structure while not destroying the data structure;
3. Find: to search and locate a specific element (usually by its numeric value or features) inside a data structure; such element may not exist in the structure in the first place.

A dictionary is implemented, in practice, commonly by two data structures called the **arrays** and the **linked lists**.



Observation: What's the minimum?

{70, 31, 9, 78, 36, 73, 50, 40, 21, 75}

Pseudocode

A "bubble sort" algorithm can be described by the following pseudocode:

```
Repeat for $n$ times ($n$ is the total number of elements):
    *Iterate* through all elements, and at the $i$-th location:
        **If** element $i$ and element ($i$+1) were out of order:
            **then** swap them
```