Arrays and Lists

C.S. Marron
cmarron@umbc.edu

CMSC 341 — Data Structures

Scenario 1: Class Roster

Roster Requirements

Store a student roster for a typical CS class (about 40 students). The data for an individual student includes first and last name, email address, UMBC student id number, and status (enrolled, waitlisted, or dropped). In addition, each student will be given a one-up class ID number (1...#students).

I need to be able to perform the following operations:

1. Create a new student entry.
2. Look-up a student by class id number.
3. Search for a student by name.
4. Change the status of a student.
5. Correct a student's information.
6. Print a roster in class id order.

Scenario 1: Class Roster

Reasonable Assumptions

What are some reasonable assumptions?

Here are a few things I came up with...

1. Use a `Student` object to store an individual student's data.
2. We will never need to store more than ≈ 100 `Student` objects.
3. We never have to delete an object, just change the status.
4. Anything else?

Scenario 1: Class Roster

Array or List?

You can use either an array (plain array, no `vector`) or a linked list.
Think about the application in terms of basic CRUD operations:

1. Create. How do we create a new student entry?
2. Retrieve. How do we access a student's information?
3. Update. How do we modify a student's data?
4. Delete. How do we remove a student entry?

In this scenario, what is the performance of these operations for An array? a linked list? Which operations are most relevant? How can you compare performance?

Scenario 2: Botnet Hosts

Host List Requirements

I am in control of a large botnet (I am evil) and need to keep a list of all the subverted hosts I have access to. For a single host, I will record its IP address, geographic information (country and region), OS, OS Version, rootkit name and version. On a daily basis, new hosts are subverted and old hosts disappear, so my inventory is in continual flux. I sell access to the botnet, and my customers usually requests hosts from specific geographic regions.

I need to be able to perform the following operations:

1. Create and delete many entries daily.
2. Search for hosts by geographic region.
3. Search or hosts by rootkit.
4. Update the information for a host.

Scenario 2: Botnet Hosts

Reasonable Assumptions

What are some reasonable assumptions?

Here are a few things I came up with...

1. Use a `Host` object to store an individual host's data.
2. The number of hosts is highly variable.
3. There's no natural order to the hosts.
4. Anything else?

Scenario 2: Botnet Hosts

Array or List?

You can use either an array (plain array, no `vector`) or a linked list.
Think about the application in terms of basic CRUD operations:

1. Create. How do we create a new host entry?
2. Retrieve. How do we access a host's information?
3. Update. How do we modify a host's data?
4. Delete. How do we remove a host entry?

In this scenario, what is the performance of these operations for An
array? a linked list? Which operations are most relevant? How can
you compare performance?

In-Class Exercise: Sparse Vectors

The Problem
In many computational applications, including machine learning and mathematical modeling, we need to store vectors of data. We mean *mathematical* vectors, not the vector class.

- ▶ Often have a very large number of variables.
- ▶ Frequently only have valid data for a small subset of the variables.
- ▶ Variables for which we have no data are called *missing values*.

Example
If a survey asks the respondents to provide demographic data (age, race, gender, etc.), many may choose not to provide this information, resulting in missing values.

Reading Assignment

Review the following in the textbook

1. Section 3.1: Using Arrays (3.1.1), Two-Dimensional Arrays (3.1.3)
2. Section 3.2: Singly Linked Lists
3. Section 3.3: Doubly Linked Lists
4. Section 3.4: Circularly Linked Lists (3.4.1)