

Lecture 4: Arrays

Ideal array:

Ask about previous use of Arrays

- You all use arrays extensively in 101 - how would you describe them?
- Answers: List, Sequence - tells part of the story!

One dimensional array:

- A sequential block of memory broken into entries
- When you get or set something from the array, for example $a[5]$, you are moving 5 spaces ahead of $a[0]$, and getting the value
 - $\text{addr}(a[i]) = \text{addr}(a[0]) + i * \text{size}(a[0])$
- Usually the syntax is $a[i]$

Differences between Arrays and Objects:

Arrays	Objects
Values are indexed by number	Usually fields are indexed by name <ul style="list-style-type: none">- But These names are changed into numbers by the compiler- In some languages numbers, those are called TUPLES
Simple formula for the location in memory of each item: $\text{addr}(a[i]) = \text{addr}(a[0]) + i * \text{size}(a[0])$	<ul style="list-style-type: none">- Because each item in an object can be of a different size, the location from the start of the object needs to be listed out in the class. Perhaps in an array itself!- TODO: Diagram
Can have an enormous size (multi-million values in an array are no problem)	In java, because each field is listed out, there is a maximum. Maybe 500? Things should be designed to probably have quite a bit less. <ul style="list-style-type: none">- Some languages (like javascript) don't have types and objects are free to be containers of larger data sets.

Similarities: Both Objects and Arrays:

- Have Random Access to all elements (next time we'll get to algorithmic complexity - we'll call this $O(1)$ or constant time)

Questions for the class:

- What operations are involved in getting the i th value of an array?
 - Addition, multiplication, fetch
- Why usually 0-based indexing?

- It makes the formula for getting the address nicer
- GREAT COMMENT FROM STUDENT: If you are representing the index by a certain number of bits, you'll have a zero (0000) available, so might as well use it.
 - In java it's an int, so we ignore the whole negative side anyway, but for tighter early systems, this is a good point.
- Does every entry in the array have to use the same number of bytes (same size)?
 - Yes
- Given how an array is arranged - if we are giving the starting location, do we know the size of the array (not java just in general).
 - Answer, no - not from just the entry size and the starting address
 - How would you remedy this?
 - One possibility: store the length elsewhere
 - Another possibility, store a "end" marker, for example, in c strings end with the null character
 - That doesn't help you too much when you want to jump ahead and you don't know the size
 - Java: placed a **header** in front of all object, a tiny slice of data, which contains the length.

Random Access

- That phrase is associated with memory
 - More generally, it's associated with Constant Time Algorithms, or $O(1)$ - more on this theory next lecture!
 - But - this includes calculations that your Computer can do that requires a fixed amount of operations.
 - Basic Operations in the machine:
 - Reading and Writing from Memory
 - Addition, Subtraction, Multiplication
 - Division and Remainder (%)
 - Bitwise Operations (and, or, xor)
 - Bit Shifts
 - Conditionals
 - When combined, there's a lot you can do!
 - More complex mathematical operations
 - More complex conditionals
- Sqrt is also considered constant time (It's an iterative algorithm, but it only needs a certain precision of bits so the maximum number of iterations is fixed)

Arrays are so fast that sometimes it's more efficient to just precompute things before it's needed and look it up in an array.

Memoization

- Similar to dynamic programming, but slightly different idea
- If you have a function that takes an integer, create an array that represents values of that function we already computed. (The index is the input, and the value of the array is the output)
- Pick some value to mean "we have not computed this yet" for fibonacci it could be -1
- What if for every time a function is called, we look to see if we've computed it. If so, return that value
 - If not, go through the algorithm, but update the array before we return

- Works if the function has no side effects
 - Can be done automatically in languages where these side effects are not allowed (functional programming language)
 - Good for “mathematical” type problems.

Arrays in Java

Show how an array is created in java

- `Int[] arrayOfInts`
- `Int[] arrayOfInts = {4, 7, 2, 45, 3}`

Arrays in Java include a header of fixed length preceding the array (which has the length and a bunch of other information, including some information contained in all java objects.)

LOOPING OVER ITEMS

In addition to the traditional way of looping, you can do it this way:

```
public class HelloWorld {
    public static void main(String[] args) {
        int[] x= {12, 3, 54, 2};

        for(int item: x) {
            System.out.println(item);
        }
    }
}
```

Each item in the array is the same type

- You could have an array of primitive `int[]`
- You could have array `Object[]` - which can contain any type of object.
 - Why does that work?
 - Every reference is the same length.

Strings in java are stored as character arrays, but they are IMMUTABLE. Get a character array:

```
char[] charArray = text.toCharArray();
```

Advanced Topics in Arrays

We want to make use of the following requirements of an array

- Random Access, Given an index `i`
- In general, we add `i * (element_size)` to the memory address at the start. This is a constant-time operation (we'll be talking about that more)

Circular Arrays

- This is a “Max Sklar” term, but I think it’s a good name for this common situation.
- Only integers from `(0, 1, 2, ..., n-1)` can access values in the array. What if we have a counter that we want to keep increasing without getting an array-out-of-bounds exception.
- Use the modulo/remainder function. This gives every integer a value.

- One of the things that annoys me about the MOD function is that it returns negative values... I'd prefer (0, 1, 2, 3, ..., n-1). But you can just add n if it's negative
- The array appears to wrap around and repeat items.

Can we decide to have skip arrays? Reverse order arrays?

- All we need is a function from i to the memory address
- Should be a "view" from a base array

Multidimensional Arrays

- In java, it stores a reference to the next array
 - Doesn't this mean that these multidimensional arrays can have rows of different lengths
 - Not contiguous in memory
- What if we wanted it to be contiguous in memory?
 - For an N x M matrix, create an array of NxM values.
 - The formula for $\text{addr}(a[i][j]) = \text{addr}(a[0][0]) + \text{element_size} * (N * i + j)$

Triangular arrays (Optional)

- Storing symmetric matrices. For example, distances between points or cities.
- n rows => $n(n + 1) / 2$ elements (derive)
- Nth row has (n + 1) items
- $\text{addr}(c[i, j]) = \text{addr}(c[0, 0]) + c.\text{size} * (i(i + 1) / 2 + j)$

Can this be generalized?

- Sure - let's say you can easily compute the function $f(i)$ which is the start of the ith row.
- $\text{addr}(c[i, j]) = \text{addr}(c[0, 0]) + c.\text{size} * (f(i) + j)$
- Also, the ith row will have $f(i + 1) - f(i)$ elements
- $f(0) = 0$, $f(i)$ must be increasing
- You CANNOT in general do this from $g(i)$ which tells us the number of elements in row i and guarantees $O(1)$ overhead and time.

Proto Hash Table

- Example: Counting Characters (large character set, small character set)
- Create an `int[]` array a of length 256 (to cover all the basic characters)
- Loop through the characters in the string, and for each character c, call `a[c] ++`.
- Now `a[c]` contains the count of any character c

[Optional Question] Can you have objects without types?

- You'd need to know the sizes, at least
 - So - standardize everything to a reference?
- An interesting language for this is Javascript - not to be confused with Java
- In Javascript, there are numbers, strings, arrays and objects
- Brenden Eich (1995) - for Netscape (Marc Andreessen)
 - Language for web browsers
 - What was the web like before javascript?
 - Capitalized on Java's Popularity at the time (marketing)
- Show them JSON, javascript, etc
 - You store the string as well as the object