CS 1112 Introduction to Computing Using MATLAB

Instructor: Dominic Diaz

Website: https://www.cs.cornell.edu/courses/cs1112/2022fa/

Today: sorting/efficiency

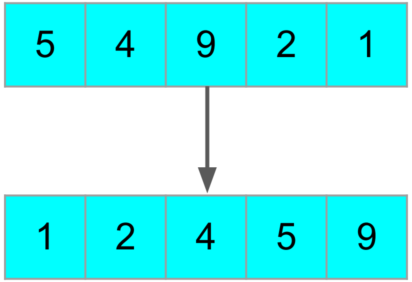
Agenda and announcements

- Last time
  - Merge sort
  - Binary search
- Today
  - An algorithm for sorting
  - Efficiency analysis
- Announcements
  - Project 6 due Dec 5th
  - Code ‘til you drop session on Dec 14th
  - Final exam on Thursday, December 15th from 2 - 4:30 PM in Olin 155
    - Check your “final exam time and location” CMS assignment if you have an SDS letter. If you have 3+ finals in a 24 hour period, submit a regrade request and we can reschedule this exam.

Searching is easier in a sorted list

Last time: binary search repeatedly halved our search space.

First we need to sort our list though.



wide at SuperPages.com 195 Car C

Cartage New England Inc 26 Allen Ln Ipswich 01938....978 356-9960
Cartagema Lydia 18 Jewett Ros 02131....617 323-7639
Cartagena Avith 9 Bancroft Rox 02119....617 442-9780
B Hyd 02136....617 361-5253
Jessica 50 Decatur Cha 02129....617 241-0152
Lucilla 174 Harvard Cam 02139....617 491-5621
M 95 Rowe Ros 02131....617 323-9713
Melvin 501 Green Cam 02139....617 576-1061
Carte Nicholas 18 Appleton Boston 02116....617 695-6996
Cartegena O 4 Milford Bos 02118....617 338-8219
Carten Thos J Sr & Claire 1 Paradise Rd Mil 02186....617 698-6163
Thomas & Kathleen 50 Thompson Ln Mil 02186....617 696-6919
Carter A Ros 02131....617 327-2257
A Roxbury....617 442-5230
A 31 Bethune Wy Roxbury 02119....617 442-1219
A 260 Putnam Av Cambridge 02139....617 492-4174
A M 255 Maschsts Av Bos 02115....617 266-7153
Adams 361 Centre St Mil 02186....617 698-9074
Alice 108 Kilmarnock Bos 02215....617 425-0193
Alice 45 Market Cambridge 02139....617 945-2711
Andrew F 62 Vinal Av Som 02143....617 625-7623
Carter Anne MD 1101 Beacon Bro 02446....617 739-1022
Carter Athens 272 Newbury Boston 02116....617 536-6329
B E 68 Gladeside Av Mat 02126....617 296-6911
Carter Barbara L MD Tufts-New England Medical Center Bos 02111 Call....617 636-0051
Carter Becky Bos 02114....617 523-4368
Bernard J 112 Gladstone E Bos 02128....617 567-3430
Bithiah 25 Medway Dor 02124....617 298-8713
Blake 26 Mt Vernon Bos 02108....617 367-9931
Carter Broadcasting Co 20 Park Plz Bos 02116....617 423-0210
Carter & Burgess Consultants Inc 23 East St Cam 02141....617 225-0200
Carter C 2000 Commwlth Av Bri 02135....617 782-2118
C 228 Faywood Av East Boston 02128....617 569-1545
C 359 Harvard Cam 02138....617 491-4822
C 610 Walk Hill Mat 02126....617 296-6392
C & M 43 Burroughs Jam 02130....617 524-9558

Carter F 24 Hillock Ros 02131....617 327-1105
Faye & Ricky 357 Columbus Av Bos 02116....617 437-7331
Francis S 134 Temple W Rox 02132....617 323-6781
Franklin & Anne 221 Mt Auburn Cam 02138....617 354-0798
Fred 42 Haverford Jam 02130....617 524-3078
Fred 96 Hinckley Rd Mil 02186....617 698-1343
G & R 8 Verdun Dor 02124....617 436-8906
G T 27 Franklin Av Som 02145....617 623-7121
Gayle 25 Frontenac Dor 02124....617 825-0322
Geo S 115 Moss Hill Rd Jam 02130....617 522-3215
George 125 Nashua Bos 02114....617 367-9548
Carter Halliday Associate 107 S Street Bos 02111....617 456-1689
Carter Harry F 26 Rusing Brk Rd W Rox 02132....617 325-5465
Carter Hide Co Inc 146 Summer Bos 02110....617 542-7987
Carter Hilary 61 Harvey Cam 02140....617 876-2750
Horace 241 Walnut Av Roxbury 02119....617 442-5307
Howard Jr 26 Notre Dme Rox 02119....617 445-5552
J Cam....617 354-2688
J 15 Chatham Bro 02446....617 232-7990
J 518 Harvard Bro 02446....617 730-9483
J 775 Vfw Pkwy West Roxbury 02132....617 323-5574
Carter J Jacques MD 1 Brookline Pl Bro 02446....617 735-8787
Carter J M 1410 Columbia Rd S Bos 02127....617 464-1040
Carter J M Ornamental Ironworks Call....Pembroke TelNo 617 436-5353
Carter J Veal Co 48 Newmarket Sq Rox 02118....617 442-1775
Carter James 1573 Cambridge St Cam 02138....617 492-1214
James 182 Fisher Av Roxbury 02120....617 739-2193
James 37 Gold Star Rd Cambridge 02140....617 876-8841
Jas L 14 Roseberry Rd Mat 02126....617 361-0773
Jane 114 Adena Rd Newton 02465....617 964-0435
Jeffrey 41 Warren Av Bos 02116....617 426-5994
John 11 Mansfield Bri 02134....617 987-2163
John 327 Summer Bos 02210....617 423-4334
John 40 Westwind Rd Dor 02125....617 282-1235
June O 329 A Summit Av Bri 02135....617 734-6109
K 38 Browning Av Dorchester 02124....617 265-8456
K 17 Esmond Dorchester 02121....617 282-1593

Carter Nella E 333 Maschsts Av Bos 02115....617 267-6483
Nicholas S F 115 Randolph Av Mil 02186....617 698-5307
Nick 21 Fairfield Bos 02116....617 267-5222
Nick & Debbi 196 Herrick Rd Newton 02459....617 527-0480
Nicole....617 698-0713
Norman G 38 Chickatawbut Dor 02122....617 822-1203
P 94 Crestwood Pk Rox 02121....617 427-4754
P E 501 E Sixth S Bos 02127....617 268-4213
P L 44 Hutchings Rox 02121....617 427-9170
P R 91 Bynner Jam 02130....617 983-8692
Paul & Constance 114 Anawan Av W Rox 02132....617 325-2036
Paul E 501 E Sixth St S Bos 02127....617 268-4546
Paul M 27 Union Bri 02135....617 787-2115
Carter Pile Driving Inc 17 Beaver Ct Framingham 01702....Wellesley TelNo 781 235-8488
Carter Prudence 46 Franklin Watertown 02172....617 393-3782
Prudence 46 Franklin Watertown 02172....617 926-7063
Reginald 106 Brunswick Dorchester 02121....617 541-2843
Renee & Andrew 10 Walnut Bos 02108....617 720-3765
Carter Rice Dowd Bulkley Dunton Publishing 163 Main Wilmington 01887
Toll Free-Dial '1' & Then....800 638-1671
Cust Svc-Industrial Prod 613 Main Wilmington
Toll Free-Dial '1' & Then....800 619-7447
Cust Svc-Printing 613 Main Wilmington
Toll Free-Dial '1' & Then....800 648-7447
Headquarters 613 Main Wilmington 01887 Call....978 988-7447
Ingalls Cronin 163 Main Wilmington 01887
Toll Free-Dial '1' & Then....800 638-1673
Carter Richard 1079 Commwlth Av Brighton 02215....617 987-0836
Richard A 97 Mt Vernon Bos 02108....617 566-7293
Carter Richard A MD 170 Commwlth Av Bos 02116....617 267-0710
Carter Richard K 15 Mercer S Bos 02127....617 268-0448
Robert L 175 Richdale Av Cam 02140....617 864-1535
Roger 150 St Botolph Bos 02115....617 424-6148
Roy 44 Concord Av Cam 02138....617 491-6115
Royce 18 Seminary Cha 02129....617 241-0418

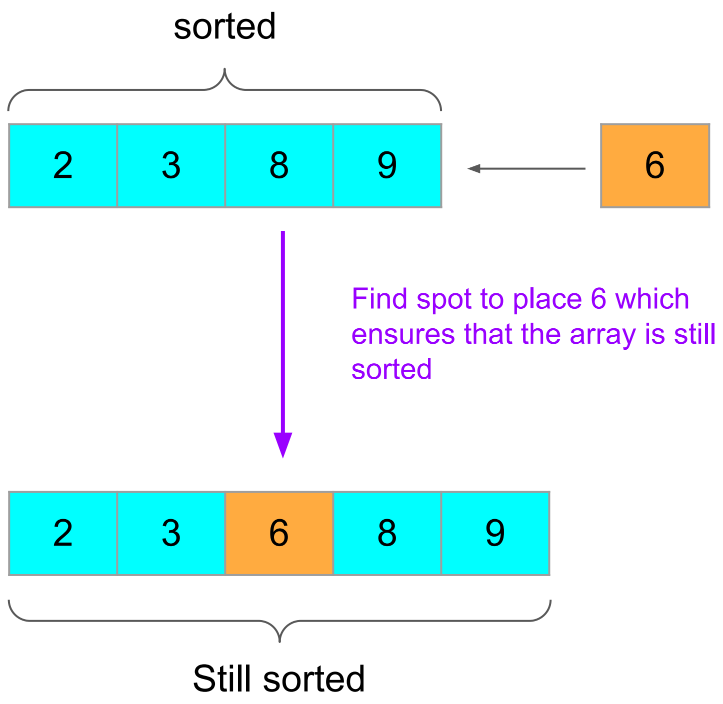
There are many algorithms for sorting

- Merge sort (discussed last lecture)
- Selection sort (exercise yesterday)
- **Insertion sort (discussed this lecture)**
- Bubble sort (read insight section 8.2)
- Quick sort (a variant used by MATLAB's built-in sort function)

- **Each has advantages and disadvantages. Some algorithms are faster (time efficient) while others are memory-efficient.**

The insertion process

Given a sorted array v, insert a number x such that the result is sorted.



Insertion process

2	3	8	9	6

Append 6 to the end of the sorted array

2	3	8	6	9

Since 9 > 6, swap

2	3	6	8	9

Since 8 > 6, swap

2	3	6	8	9

Since 3 < 6, stop swapping

Array is sorted once again!

Another insertion

2	3	6	8	9

Insert

-1

2	3	6	8	9	-1	Append -1 to the end
2	3	6	8	-1	9	Since 9 > -1, swap
2	3	6	-1	8	9	Since 8 > -1, swap
2	3	-1	6	8	9	Since 6 > -1, swap
2	-1	3	6	8	9	Since 3 > -1, swap
-1	2	3	6	8	9	Since 2 > -1, swap

Sorted again, yay!

Sort vector x using the Insertion Sort algorithm

5	7	-1	2	4

To sort using insertion sort, need to start with a sorted subvector. How do you find one?

Length 1 vector is always sorted

5

Append 7 and swap until sorted again

5	7

Append -1 and swap until sorted again

5	7	-1

Append 2 and swap until sorted again

-1	5	7	2

Append 4 and swap until sorted again

-1	2	5	7	4

Take a second to think about when insertion sort is most efficient and least efficient. What should the starting vector look like?

How much “work” is insertion sort?

- In the worst case, make k comparisons to insert an element into a sorted array of k elements. To sort an array of length N, how many comparisons do we need to do?

10	7	5	3	1	-10

$$1 + 2 + 3 + \ldots + (N-1) = N(N-1)/2$$

- In the best case, make 1 comparison after inserting each new number. To sort an array of length N, how many comparisons do we need to do?

-1	4	7	8	9	10

$$1 + 1 + 1 + \ldots + 1 = N-1$$

When talking about the time complexity of an algorithm, We take the number of computations (or comparisons) in the worst case and get rid of constants: $O(N^2)$.

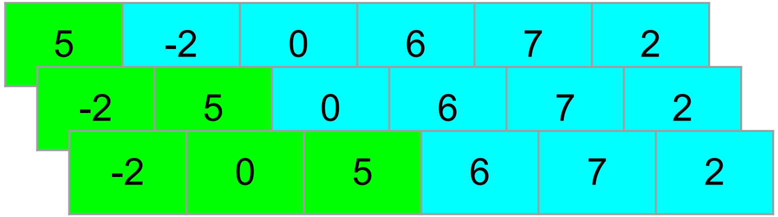
This big O corresponds to the limiting behavior… what the function tends to as N gets very large.

Efficiency considerations

- Worst case, best case, average case number of calculations (or comparisons)
- Memory use and access
    - For example, in insertion sort we can create an extra vector to store the "sorted so far" vector

Original vector	5	-2	0	6	7	2
Sorted so far vector	-2	0	5	6		

    - We could sort the vector "in place" as well (this would not incur extra memory costs)



```matlab
function x = insertSortInplace(x)
% Sort vector x in ascending order using Insertion sort
n = length(x);
for i = 1:n-1
    % Sort x(1:i+1) given that x(1:i) is sorted
    j = i;
    need2swap = x(j+1) < x(j);









end
```

```
function x = insertSortInplace(x)
% Sort vector x in ascending order using Insertion sort
n = length(x);
for i = 1:n-1
    % Sort x(1:i+1) given that x(1:i) is sorted
    j = i;
    need2swap = x(j+1) < x(j);
    while need2swap
        temp = x(j);
        x(j) = x(j+1);
        x(j+1) = temp;
        j = j-1;
        need2swap = j>0 && x(j+1)<x(j);
    end
end
```

Efficiency:

Time complexity: In the worst case, do about n*(n-1)/2 comparisons so the time complexity is O(n^2).

Space complexity: In the worst case, we need 5 *extra* spaces of memory. Constants become 1 so we say the space complexity is O(1).

Because it does not depend on n.

What is the time and space complexity of this function if the input A is a matrix of size nxn.

```
function B = pos(A)

B = zeros(size(A));
[nr, nc] = size(A);
for i = 1:nr
    for j = 1:nc
        if A(i,j) < 0
            B(i,j) = A(i,j)*(-1);
        else
            B(i,j) = A(i,j);
        end
    end
end

end
```

Time: In the code, we're doing n^2 things (n^2 comparisons) → O(n^2)

Space: We're creating n^2 + 4 extra things in memory (B, nr, nc, i, and j) → O(n^2).

When doing `a*n^2 + b*n + c` computations or creating `a*n^2 + b*n + c` extra things in memory, the complexity is O(n^2).

Get rid of constants and lower order terms.

Extra slide on complexity

When your code takes an input of length n, how many computations (or comparisons) does it do? How much *extra* memory space does your code require (other than the length n input)?

Steps:

1. Compute the number of computations (or extra spaces of memory) to get a function in terms of n. For example, 10*n^3 + n + 10.
2. Constants in this function become 1 and we throw away lower order terms. In the example above, we would say the time (or space) complexity is O(n^3).
    a. If we do a constant number of computations or only constant memory space is used (constant means that these numbers don't grow with n), we say the time (or space) complexity is O(1).

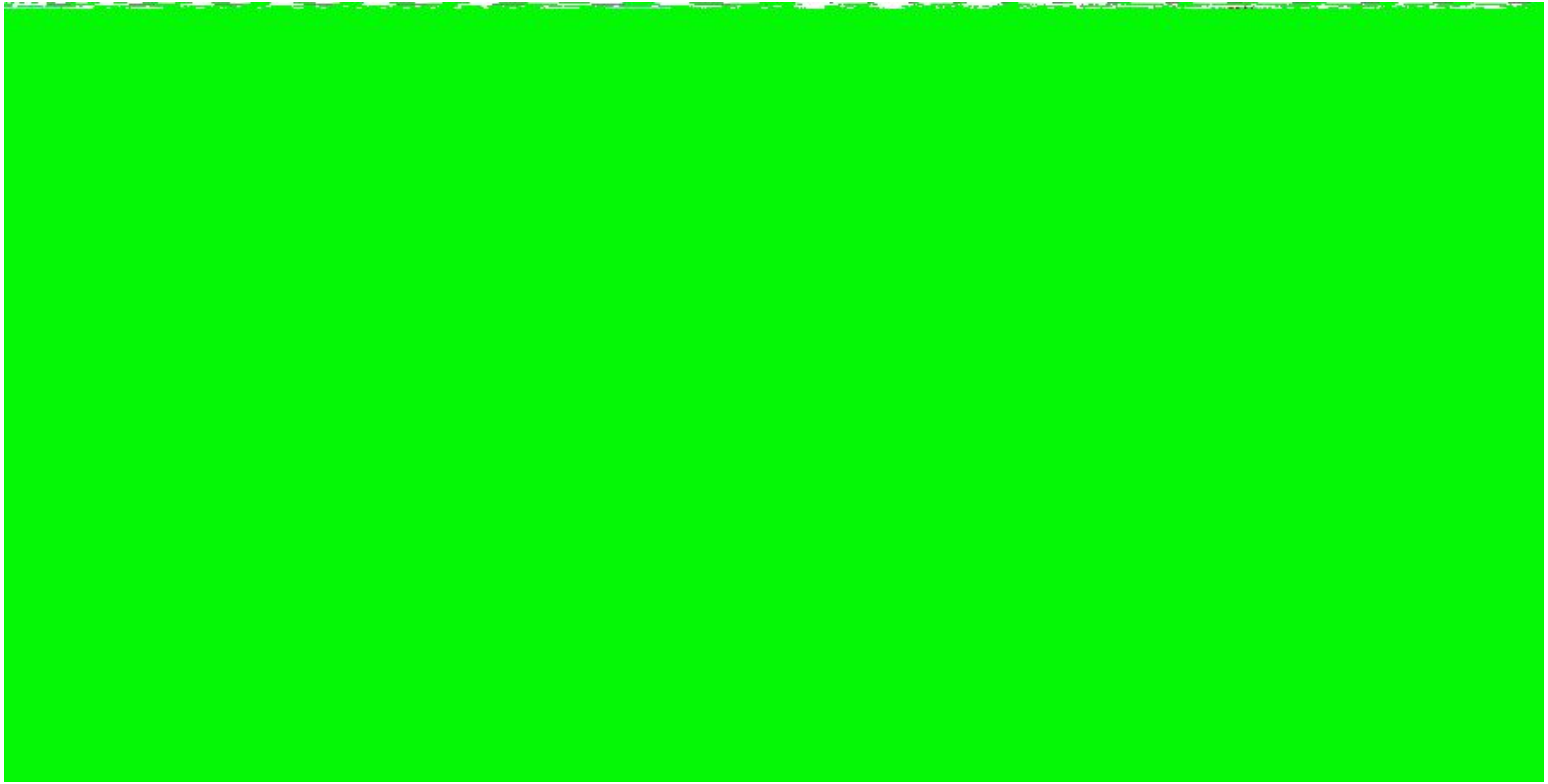
What does this "Big-O" mean? It means "what is the limiting behavior of this function?". When we take the limit as n gets large what term dominates? We don't care about the constant but we care about degree of the largest term. This is a very useful tool when comparing algorithms.

How can we use complexity to compare algorithms?

Consider the two following codes that do the exact same thing.

<pre>% Given some 2D array A of size nxn
for i = 1:size(A,1)
     for j = 1:size(A,2)
          if i == 1
               A(i,j) = Inf;
          end
     end
end</pre>
	<pre>% Given some 2D array A of size nxn
	for j = 1:size(A,2)
	     A(1,j) = Inf;
	end</pre>

The left code does n^2 comparisons plus n variable assignments. So we would say the time complexity of the left code is O(n^2). The right code does n assignments. So we would say the time complexity of the right code is O(n). The right code is much faster and we are able to use this complexity analysis to compare these algorithms that do the same thing.



Check out this link for a comparison of different sorting algorithms:
https://www.toptal.com/developers/sorting-algorithms

What do I need to know in terms of sorting and searching?

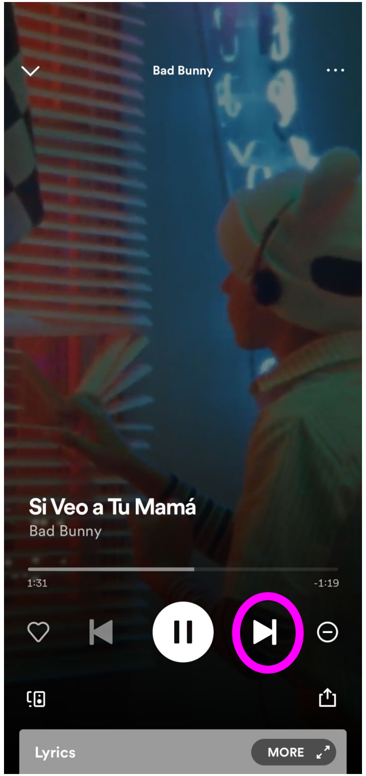
- Should be able to implement a sorting algorithm from pseudocode (like this week's exercise)
- Should be able to analyze the efficiency of a sorting algorithm (time and space complexity in the worst case scenario).
  - Merge sort is a little too difficult to analyze for this course
  - Practice with insertion sort from ex 14
  - Practice with bubble sort from the textbook
- Should be able to complete an incomplete binary search code.

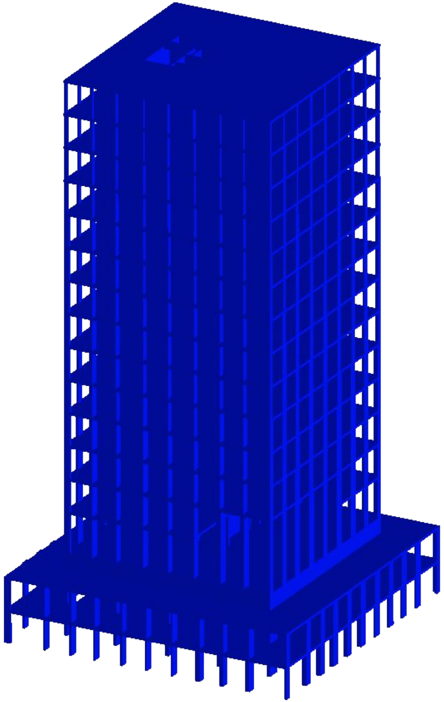
What we learned in this course

- Develop and implement algorithms for solving problems
- Develop programming skills
  - Design, implement, document (comments), test, and debug
- Programming tool bag
  - Functions for reducing redundancy
  - Control flow (if-else, loops)
  - Data structures, types
  - Graphics
  - File handling
- Applications and concepts
  - Image processing
  - Object-oriented programming
  - Sorting and searching
  - Randomness
  - Simulation, sensitivity analysis
  - Computational efficiency

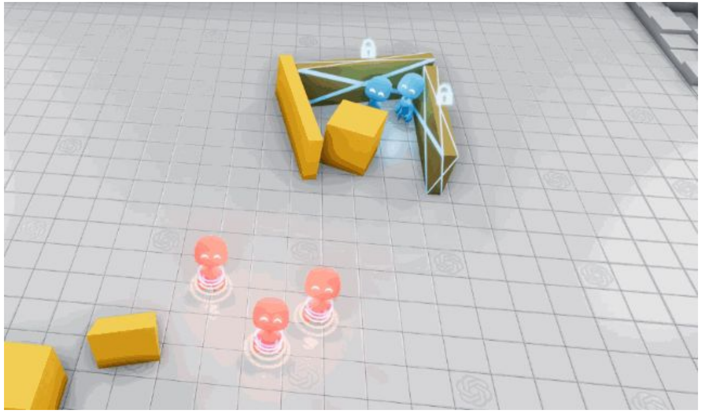
Where to go from here?

- Learn another programming language
- Explore cool/impactful applications!

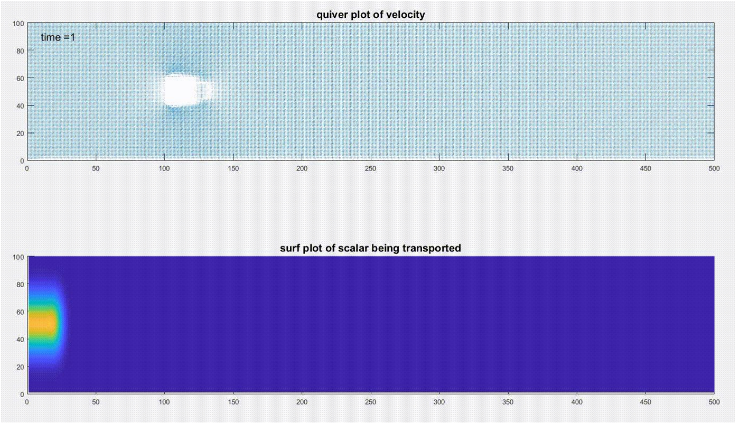




https://www.appliedscienceint.com/structural-engineering-services/



https://openai.com/blog/emergent-tool-use/



Computing gives us insight into a problem

- We build models and write programs so that we can “play” with the models and parameters
- Good programs…
  - Have been thoroughly tested
  - Are cleanly organized
  - Are well-documented
  - Use appropriate data structures and algorithms
  - Are efficient in time and memory

Best wishes and good luck with your exams!!