- Previous Lecture:
  - Working with images
- Today's Lecture:
  - Characters and strings
- Announcements:
  - Prelim 2 will be returned at end of lecture. If your paper isn't here, pick it up from CSIII2 consultants in ACCEL during consulting hrs (starting today after 4pm)
  - Discussion this week in classrooms as listed on roster
  - Project 4 posted. Due Mon, Apr 4th, at 11pm

## Characters & strings

- We have used strings already:
  - n = input('Next number: ')
  - sprintf('Answer is %d', ans)
- A string is made up of individual characters, so a string is a I-d array of characters
- 'CS1112 rocks!' is a character array of length 13; it has 7 letters, 4 digits, 1 space, and 1 symbol.

Can have 2-d array of characters as well

# Matlab types: char, double, uint8, logical There is not a type "string"! What we call a string is a <u>1-d array of chars</u> a is a 1-d array with type char components. We call a a "string" or "char array" a 'C'S'1' b is a 1-d array with type double $b = [3 \ 9]$ components. double is the default type for numbers in Matlab. We call **b** a "numeric array" c is a I-d array with type uint8 c = uint8(b)components. We call c a "uint8 array" ${\tt d}$ is a scalar of the type logical. We call ${\tt d}$ d = rand > .5a "boolean value"

# Strings are important in computation

Numerical data is often encoded in strings. E.g., a file containing Ithaca weather data begins with the string

W07629N4226

meaning

Longitude: 76° 29' West Latitude: 42° 26' North

We may need to grab hold of the substring w07629, convert 076 and 29 to the numeric values 76 and 29, and do some computation

Lecture 17

# Comparison of genomic sequences is another example of string computation

- E.g., looking for a pattern:
   Given the sequence ATTCTGACCTCGATC...
   Look for the pattern ACCT
- E.g., quantifying the difference between sequences:

# ATTCTGACCTCGATC ATTCGTGACCTCGAT What if this nucleotide is removed?

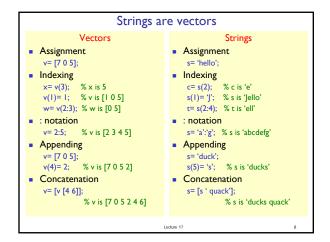
## Single quotes enclose strings in Matlab

Anything enclosed in single quotes is a string (even if it looks like something else)

- 100' is a character array (string) of length 3
- 100 is a numeric value
- 'pi' is a character array of length 2
- pi is the built-in constant 3.1416...
- $\mathbf{x}'$  is a character (vector of length I)
- x may be a variable name in your program

ecture 17

Lecture slides 1



```
Some useful string functions

str= 'Cs 1112';

length(str) % 7

isletter(str) % [1 1 0 0 0 0 0]

isspace(str) % [0 0 1 0 0 0 0]

lower(str) % 'cs 1112'

upper(str) % 'CS 1112'

ischar(str)

% Is str a char array? True (1)

strcmp(str(1:2),'cs')

% Compare strings str(1:2) & 'cs'. False (0)

strcmp(str(1:3),'Cs')

% False (0)
```

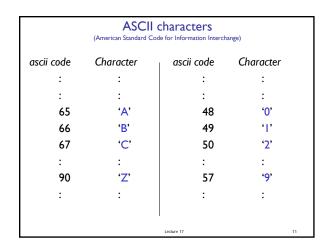
```
Example: capitalize Ist letter

Write a function to capitalize the first letter of each word in a string. Assume that the string has lower case letters and blanks only. (OK to use built-in function upper)

function [str, nCaps] = caps(str)
% Post: Capitalize first letter of each word.
% str = partially capitalized string
% nCaps = no. of capital letters
% Pre: str = string with lower case letters & blanks only

look for the spaces
Look For The Spaces

See caps.m
```



```
Character vs ASCII code

str= 'Age 19'
%a 1-d array of characters
code= double(str)
%convert chars to ascii values
str1= char(code)
%convert ascii values to chars
```

```
Arithmetic and relational ops on characters

'c'-'a' gives 2
'6'-'5' gives I

letter1='e'; letter2='f';

letter1-letter2 gives -I

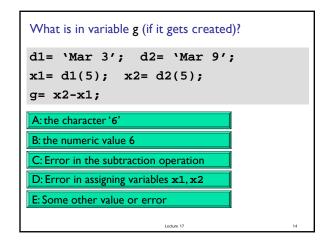
'c'>'a' gives true

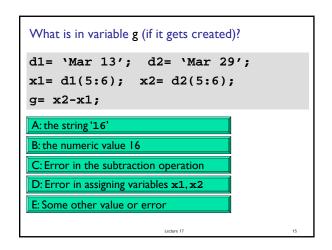
letter1==letter2 gives false

'A' + 2 gives 67

char('A'+2) gives 'C'
```

Lecture slides 2





Lecture slides 3