Introduction & MATLAB
Digital Geometry Processing

- Focus: what can we do with a triangle mesh?

Curvature

Gradient
Digital Geometry Processing

Focus: what can we do with a triangle mesh?
Digital Geometry Processing

• 3 homework assignments and a final project
  – HW1 – 29/03
  – HW2 – 22/04
  – HW3 – 06/05
  – Final project – last two lectures

• Submission in pairs or singles
MATLAB

- MATLAB is a computing environment that is especially advantageous for matrix manipulations and data analysis
  - Matrix manipulations are very efficient
  - Displaying graphs, images and 3D meshes requires only a few lines of code
Matrix Operations

- Many built in functions, use them as much as possible
- Standard operators: + - * / ^
- A dot before the operator makes it elementwise
- The backslash \ operator solves linear systems!
## Matrix Operations

```
>> a = [1:3; 10,11,12; 20:5:30]
   a =
       1     2     3
       10    11    12
       20    25    30

>> b = [1; 2; 3]
   b =
       1
       2
       3

>> a*b
   ans =
       14
       68
       160
```

- **a** contains all integers from 1 to 3 (row vector).
- **b** is an integer row vector from 1 to 3.
- **a** is a matrix containing integers from 20 to 30, increment of 5.
- Matrix multiplication:
  - `ans` is the result of multiplying matrix **a** with row vector **b**.
Matrix Operations

Row vector, at iteration i the value of j is the i-th entry of the vector

Supress output

1-based indices

for j = 1:5
    b = a*b;
end

b =
47375118
211247028
509995440

b(1) = 3;

b =
3
211247028
509995440
Matrix Operations

• Use matrix operations whenever you can, this is what MATLAB is meant for.
• If you write everything with loops and indices, you will spend a lot of time waiting.
Matrix Operations

• Sparse matrices are awesome, use them when you do not have many nonzero entries.

• Useful functions: `speye`, `spdiags`, `sparse`, `full`

• `bsxfun` is another VERY useful function
Matrix Operations

>> Asparse = spdiags((1:1e4)', 0, 1e4,1e4);
>> Afull = full(Asparse);
>> b = rand(1e4,1);

>> tic; Afull*b; toc
Elapsed time is 0.214862 seconds.

>> tic; Asparse*b; toc
Elapsed time is 0.000111 seconds.

A row vector is transposed to get a column vector

1e4 rows, 1 column of random values

Time from last tic
Scripts

- .m file extension (all MATLAB code files)
- You can run scripts directly
- Debugging is easy
Functions

• Usually each function is written in a separate file, the file name should be identical to the (first) function name

• You can define a few functions in a file, but other files will only be able to access the first one
Performance

• I tried, and I tried, I cannot find a built in MATLAB function that does what I need, and the code is really slow... what can I do?
  – Try MATLAB’s profiler (profile clear, profile on, profile viewer)
  – Sometimes it’s worth writing C/C++ code and run it from MATLAB (mex)
Classes

• Classes are less common in MATLAB, but can be very useful

• Allow passing arguments by reference (try `doc classdef` and `doc handle`)