COT 6936: Topics in Algorithms

Giri Narasimhan

ECS 254A / EC 2474; Phone x3748; Email: giri@cs.fiu.edu
HOMEPAGE: http://www.cs.fiu.edu/~giri
https://moodle.cis.fiu.edu/v2.1/course/view.php?id=612

Mar 6, 2014

Presentation Outline

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

LEDA Small Example

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

Algorithm 1 Planarity Testing

1: #include ⟨LEDA/graph/graph_alg.h⟩

LEDA Small Example

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

Algorithm 2 Planarity Testing

1: #include ⟨LEDA/graph/graph_alg.h⟩

2: using namespace leda;

LEDA Small Example

Algorithm 3 Planarity Testing

1: #include ⟨LEDA/graph/graph_alg.h⟩

2: using namespace leda;

3: int main(int argc, char *argv[])

LEDA Small Example

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

Algorithm 4 Planarity Testing

1: #include ⟨LEDA/graph/graph_alg.h⟩

2: using namespace leda;

3: int main(int argc, char *argv[])

4: {

    5: graph G;

LEDA Small Example

Algorithm 5 Planarity Testing

1: #include ⟨LEDA/graph/graph_alg.h⟩

2: using namespace leda;

3: int main(int argc, char *argv[])

4: {

  5: graph G;

  6: string filename(argv[1]);

LEDA Small Example

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

Algorithm 6 Planarity Testing

1: #include ⟨LEDA/graph/graph_alg.h⟩

2: using namespace leda;

3: int main(int argc, char *argv[])

4: {

  5: graph G;

  6: string filename(argv[1]);

  7: G.read(filename);

LEDA Small Example

Algorithm 7 Planarity Testing

1: #include ⟨LEDA/graph/graph_alg.h⟩

2: using namespace leda;

3: int main(int argc, char *argv[])

4: {

    5: graph G;

    6: string filename(argv[1]);

    7: G.read(filename);

    8: cout ≪ PLANAR(G) ≪ endl;

LEDA Small Example

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

Algorithm 8 Planarity Testing

1: #include ⟨LEDA/graph/graph_alg.h⟩

2: using namespace leda;

3: int main(int argc, char *argv[])

4: {

    5: graph G;

    6: string filename(argv[1]);

    7: G.read(filename);

    8: cout ≪ PLANAR(G) ≪ endl;

9: }

Presentation Outline

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

LEDA Overview

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

- A C++ library of data structures and algorithms
- Efficient code
- Amply tested
- Extensively documented
- Widely installed

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

LEDA Overview

- **Basic Data Types** float, bigfloat, rational, real, list, map, queue, stack, array, vector, set, matrix, polynomials, string, . . .

LEDA Overview

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

- **Basic Data Types** float, bigfloat, rational, real, list, map, queue, stack, array, vector, set, matrix, polynomials, string, . . .

- **More Data Types** trees, balanced trees, priority queues, heaps, binomial heaps, Fibonacci heaps, dictionary, hash table, string, tuple, . . .

LEDA Overview

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

- **Basic Data Types** float, bigfloat, rational, real, list, map, queue, stack, array, vector, set, matrix, polynomials, string, . . .

- **More Data Types** trees, balanced trees, priority queues, heaps, binomial heaps, Fibonacci heaps, dictionary, hash table, string, tuple, . . .

- **Basic Algorithms** Sorting, Searching, Hashing, alignment, partition, . . .

LEDA Overview

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

- **Basic Data Types** float, bigfloat, rational, real, list, map, queue, stack, array, vector, set, matrix, polynomials, string, . . .

- **More Data Types** trees, balanced trees, priority queues, heaps, binomial heaps, Fibonacci heaps, dictionary, hash table, string, tuple, . . .

- **Basic Algorithms** Sorting, Searching, Hashing, alignment, partition, . . .

- **Graph Algorithms** BFS, DFS, MST, Dijkstra, Floyd-Warshall, maxflow, mincut, matching, isomorphism, markov chain, planar graphs, planarity testing, graph drawing algorithms, triangulations, . . .

- **Geometric Data Structures and Algorithms** point, line, ray, segment, polygon, circle, plane, rectangles, triangle, hull, plane sweep, delaunay trinagulations, Voronoi diagrams, range trees, interval trees, kd-trees, point locations, segment trees, transformations, . . .

LEDA Overview Cont'd

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

- **Geometric Data Structures and Algorithms** point, line, ray, segment, polygon, circle, plane, rectangles, triangle, hull, plane sweep, delaunay trinagulations, Voronoi diagrams, range trees, interval trees, kd-trees, point locations, segment trees, transformations, . . .

- **Linear Algebra** modular arithmetic, long integers, number theory algorithms, . . .

LEDA Overview Cont'd

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

- **Geometric Data Structures and Algorithms** point, line, ray, segment, polygon, circle, plane, rectangles, triangle, hull, plane sweep, delaunay trinagulations, Voronoi diagrams, range trees, interval trees, kd-trees, point locations, segment trees, transformations, . . .

- **Linear Algebra** modular arithmetic, long integers, number theory algorithms, . . .

- **Graph Types and Operations** undirected, directed, weighted, planar, adding/removing edges/vertices, faces, . . .

LEDA Overview Cont'd

- **Geometric Data Structures and Algorithms** point, line, ray, segment, polygon, circle, plane, rectangles, triangle, hull, plane sweep, delaunay trinagulations, Voronoi diagrams, range trees, interval trees, kd-trees, point locations, segment trees, transformations, . . .

- **Linear Algebra** modular arithmetic, long integers, number theory algorithms, . . .

- **Graph Types and Operations** undirected, directed, weighted, planar, adding/removing edges/vertices, faces, . . .

    - GRAPH⟨vtype, etype⟩:

- **Graphics** windows, colors, scenes, panels, menu, pixmaps, bitmaps, . . .

- **Geometric Data Structures and Algorithms** point, line, ray, segment, polygon, circle, plane, rectangles, triangle, hull, plane sweep, delaunay trinagulations, Voronoi diagrams, range trees, interval trees, kd-trees, point locations, segment trees, transformations, . . .

- **Linear Algebra** modular arithmetic, long integers, number theory algorithms, . . .

- **Graph Types and Operations** undirected, directed, weighted, planar, adding/removing edges/vertices, faces, . . .
  - GRAPH⟨vtype, etype⟩:

- **Graphics** windows, colors, scenes, panels, menu, pixmaps, bitmaps, . . .
  - graph-windows:

- **Control Structures**
  - forall_nodes(v, G)

- **Control Structures**
  - forall_nodes(v, G)
  - forall_edges(e, G)

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

LEDA Overview Cont'd

- **Control Structures**
    - forall_nodes(v, G)
    - forall_edges(e, G)
    - forall_adj_nodes(v, w)
    - forall_adj_edges(e, w)

- **Control Structures**
    - forall_nodes(v, G)
    - forall_edges(e, G)
    - forall_adj_nodes(v, w)
    - forall_adj_edges(e, w)
    - forall_out_edges(e, w)
    - forall_in_edges(e, w)

- **Control Structures**
  - forall_nodes(v, G)
  - forall_edges(e, G)
  - forall_adj_nodes(v, w)
  - forall_adj_edges(e, w)
  - forall_out_edges(e, w)
  - forall_in_edges(e, w)

Presentation Outline

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

1   LEDA: Library of Efficient Data Structures and Algorithms

2   Features of LEDA

3   Another LEDA Example

LEDA Example

Algorithm 9 Planarity Testing

#include ⟨LEDA/graph/graph_alg.h⟩
using namespace leda;
int main(int argc, char *argv[])
{
    graph G;
    string filename(argv[1]); G.read(filename);
    list⟨edge⟩ edge_list;

LEDA Example

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

Algorithm 10 Planarity Testing

```
#include ⟨LEDA/graph/graph_alg.h⟩
using namespace leda;
int main(int argc, char *argv[])
{
    graph G;
    string filename(argv[1]); G.read(filename);
    list⟨edge⟩ edge_list;
    if (PLANAR(G, edge_list) == 0)
```

LEDA Example

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

Algorithm 11 Planarity Testing

```
#include ⟨LEDA/graph/graph_alg.h⟩
using namespace leda;
int main(int argc, char *argv[])
{
    graph G;
    string filename(argv[1]); G.read(filename);
    list⟨edge⟩ edge_list;
    if (PLANAR(G, edge_list) == 0)
    forall (x,edge_list) G.print_edge(x);
```

LEDA Example

Algorithm 12 Planarity Testing

```
#include ⟨LEDA/graph/graph_alg.h⟩
using namespace leda;
int main(int argc, char *argv[])
{
    graph G;
    string filename(argv[1]); G.read(filename);
    list⟨edge⟩ edge_list;
    if (PLANAR(G, edge_list) == 0)
    forall (x,edge_list) G.print_edge(x);
}
```
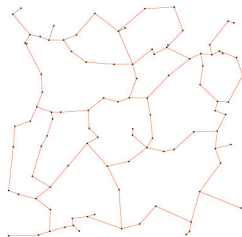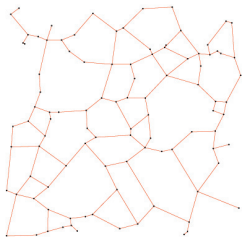

Sample LEDA Figures

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

- You can run it on any of the unix-based machines in SCIS (e.g., jaguar, ermine, stoat, . . .)

- You can run it on any of the unix-based machines in SCIS (e.g., jaguar, ermine, stoat, ...)
- Modify `.cshrc` (or other shell startup config file):
  ```
  alias add2path 'if ("$path:q" !   *"$"* ) set path=( $path $ )'
  setenv LEDAROOT /depot/LEDA-6
  add2path $LEDAROOT/Manual/cmd
  setenv LEDA $LEDAROOT/incl/LEDA
  ```

Running LEDA

- You can run it on any of the unix-based machines in SCIS (e.g., jaguar, ermine, stoat, . . .)
- Modify `.cshrc` (or other shell startup config file):
  ```
  alias add2path 'if ("$path:q" !   *"$"* ) set path=( $path $ )'
  setenv LEDAROOT /depot/LEDA-6
  add2path $LEDAROOT/Manual/cmd
  setenv LEDA $LEDAROOT/incl/LEDA
  ```
  and add `$LEDAROOT` to the env variable `LD_LIBRARY_PATH`

Running LEDA

- You can run it on any of the unix-based machines in SCIS (e.g., jaguar, ermine, stoat, . . .)
- Modify `.cshrc` (or other shell startup config file):

  ```
  alias add2path 'if ("$path:q" !   *"$"* ) set path=( $path $ )'
  setenv LEDAROOT /depot/LEDA-6
  add2path $LEDAROOT/Manual/cmd
  setenv LEDA $LEDAROOT/incl/LEDA
  ```

  and add `$LEDAROOT` to the env variable `LD_LIBRARY_PATH`

- Next, either login again or type `source .cshrc` to force the shell to read the changes to the config file.

Running LEDA

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

- You can run it on any of the unix-based machines in SCIS (e.g., jaguar, ermine, stoat, . . .)
- Modify `.cshrc` (or other shell startup config file):
  ```
  alias add2path 'if ("$path:q" !   *"$"* ) set path=( $path $ )'
  setenv LEDAROOT /depot/LEDA-6
  add2path $LEDAROOT/Manual/cmd
  setenv LEDA $LEDAROOT/incl/LEDA
  ```
  and add `$LEDAROOT` to the env variable `LD_LIBRARY_PATH`
- Next, either login again or type `source .cshrc` to force the shell to read the changes to the config file.
- Compile the `.c` file using the include flags `$LEDAROOT/incl`.

Running LEDA

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

- You can run it on any of the unix-based machines in SCIS (e.g., jaguar, ermine, stoat, . . .)
- Modify `.cshrc` (or other shell startup config file):

```
alias add2path 'if ("$path:q" !  *"$"* ) set path=( $path $ )'
setenv LEDAROOT /depot/LEDA-6
add2path $LEDAROOT/Manual/cmd
setenv LEDA $LEDAROOT/incl/LEDA
```

  and add `$LEDAROOT` to the env variable `LD_LIBRARY_PATH`

- Next, either login again or type `source .cshrc` to force the shell to read the changes to the config file.
- Compile the `.c` file using the include flags `$LEDAROOT/incl`. For e.g.,

```
g++ -O3 -I$LEDAROOT/incl -c test.c
```

Running LEDA

- You can run it on any of the unix-based machines in SCIS (e.g., jaguar, ermine, stoat, . . .)
- Modify `.cshrc` (or other shell startup config file):

```
alias add2path 'if ("$path:q" !  *"$"* ) set path=( $path $ )'
setenv LEDAROOT /depot/LEDA-6
add2path $LEDAROOT/Manual/cmd
setenv LEDA $LEDAROOT/incl/LEDA
```

  and add `$LEDAROOT` to the env variable `LD_LIBRARY_PATH`

- Next, either login again or type `source .cshrc` to force the shell to read the changes to the config file.
- Compile the `.c` file using the include flags `$LEDAROOT/incl`. For e.g.,

```
g++ -O3 -I$LEDAROOT/incl -c test.c
```

- Link and load and create executable using appropriate flags to load libraries

Running LEDA

- You can run it on any of the unix-based machines in SCIS (e.g., jaguar, ermine, stoat, . . .)
- Modify `.cshrc` (or other shell startup config file):

  ```
  alias add2path 'if ("$path:q" !   *"$"* ) set path=( $path $ )'
  setenv LEDAROOT /depot/LEDA-6
  add2path $LEDAROOT/Manual/cmd
  setenv LEDA $LEDAROOT/incl/LEDA
  ```

  and add `$LEDAROOT` to the env variable `LD_LIBRARY_PATH`
- Next, either login again or type `source .cshrc` to force the shell to read the changes to the config file.
- Compile the `.c` file using the include flags `$LEDAROOT/incl`. For e.g.,

  ```
  g++ -O3 -I$LEDAROOT/incl -c test.c
  ```

- Link and load and create executable using appropriate flags to load libraries as shown below

  ```
  g++ -I$LEDAROOT/incl -o test test.o -L$(LEDAROOT) -lleda -lm -lX11
  ```

Running LEDA

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

- You can run it on any of the unix-based machines in SCIS (e.g., jaguar, ermine, stoat, . . .)
- Modify `.cshrc` (or other shell startup config file):
  ```
  alias add2path 'if ("$path:q" !    *"$"* ) set path=( $path $ )'
  setenv LEDAROOT /depot/LEDA-6
  add2path $LEDAROOT/Manual/cmd
  setenv LEDA $LEDAROOT/incl/LEDA
  ```
  and add `$LEDAROOT` to the env variable `LD_LIBRARY_PATH`
- Next, either login again or type `source .cshrc` to force the shell to read the changes to the config file.
- Compile the `.c` file using the include flags `$LEDAROOT/incl`. For e.g.,
  ```
  g++ -O3 -I$LEDAROOT/incl -c test.c
  ```
- Link and load and create executable using appropriate flags to load libraries as shown below
  ```
  g++ -I$LEDAROOT/incl -o test test.o -L$(LEDAROOT) -lleda -lm -lX11
  ```
- Finally you are ready to execute the executable `test`

Running LEDA

COT 6936:
Topics in
Algorithms

Giri
Narasimhan

LEDA: Library
of Efficient
Data
Structures and
Algorithms

Features of
LEDA

Another LEDA
Example

- You can run it on any of the unix-based machines in SCIS (e.g., jaguar, ermine, stoat, ...)
- Modify `.cshrc` (or other shell startup config file):

```
alias add2path 'if ("$path:q" !   *"$"* ) set path=( $path $ )'
setenv LEDAROOT /depot/LEDA-6
add2path $LEDAROOT/Manual/cmd
setenv LEDA $LEDAROOT/incl/LEDA
```

  and add `$LEDAROOT` to the env variable `LD_LIBRARY_PATH`
- Next, either login again or type `source .cshrc` to force the shell to read the changes to the config file.
- Compile the `.c` file using the include flags `$LEDAROOT/incl`. For e.g.,

```
g++ -O3 -I$LEDAROOT/incl -c test.c
```

- Link and load and create executable using appropriate flags to load libraries as shown below

```
g++ -I$LEDAROOT/incl -o test test.o -L$(LEDAROOT) -lleda -lm -lX11
```

- Finally you are ready to execute the executable `test`
- Sample `Makefile` available on your class moodle site.