10.555 Bioinformatics

Principles, Methods, Applications
GenBank Statistics

Figure from: http://www.ncbi.nlm.nih.gov/Genbank/genbankstats.html
Glossaries

**Glossary of Genetics Terms:**
http://www.nhgri.nih.gov/DIR/VIP/Glossary/pub_glossary.cgi

**Glossary of Computer Science Terms:**
http://foldoc.doc.ic.ac.uk/foldoc/index.html

**Another Glossary of Genetics Terms:**
http://www.bis.med.jhmi.edu/Dan/DOE/prim6.html

**A Hypermedia Glossary of Genetics Terms:**
http://www.weihenstephan.de/~schlind/genglos.html

**An Interactive Glossary of Latin Terms:**
http://lysy2.archives.nd.edu/cgi-bin/words.exe

**Multilingual Glossary of Technical and Popular Medical Terms in 9 European Languages**
http://allserv.rug.ac.be/~rvdstich/eugloss/welcome.html
Database Link

dna, protein, est, individual genomes, gene expression data etc. from all over the world. 11,101,066,288 bases in 10,106,023 sequence records as of February 2001

PIR  http://www-nbrf.georgetown.edu/pir/
Collaboration btw the Protein Information Resource (PIR), the Munich Information Center for Protein Sequences (MIPS) and the Japanese International Protein Sequence Database (JIPID). A comprehensive, annotated, and non-redundant protein sequence database in which entries are classified into family groups and alignments of each group are available. Current Release 67.03, February 16, 2001, Contains 210045 Entries.

Swiss-Prot + TrEMBL  http://www.expasy.ch/sprot
21-Feb-2001: 93,408 entries / TrEMBL: 376,043 entries

PROSITE  http://www.expasy.ch/prosite
1040 documentation entries that describe 1386 different patterns, rules and profiles/ matrices

PDB / RCSB  http://www.rcsb.org/pdb/
12,514 structures as of June 13, 2000

PDB select  http://www.sander.embl-heidelberg.de/pdbsel/

GDB  http://gdbwww.gdb.org/
the official central repository for genomic mapping data resulting from the Human Genome Initiative

PRODOM  http://protein.toulouse.inra.fr/prodom.html
An automatic compilation of homologous domains: 174952 families as of October 1999
Database Links (CONT.)

**PROTOMAP** [http://www.protomap.cs.huji.ac.il/](http://www.protomap.cs.huji.ac.il/)
An exhaustive classification of all proteins in the swissprot database into clusters of related proteins.

Clusters of Orthologous Groups of proteins (COGs) were delineated by comparing protein sequences encoded in 34 complete genomes, representing 26 major phylogenetic lineages. Each COG consists of individual proteins or groups of paralogs from at least 3 lineages and thus corresponds to an ancient conserved domain.

**GOLD** [http://wit.integratedgenomics.com/GOLD/](http://wit.integratedgenomics.com/GOLD/)
A World Wide Web resource for comprehensive access to information regarding complete and ongoing genome projects around the world.

**BLOCKS** [http://www.blocks.fhcrc.org/](http://www.blocks.fhcrc.org/)
4071 blocks representing 998 groups documented in InterPro 1.0 keyed to SWISS-PROT 38.

**PFAM** [http://www.blocks.fhcrc.org/](http://www.blocks.fhcrc.org/)
A large collection of multiple sequence alignments and hidden Markov models covering many common protein domains. Version 5.3 of Pfam (May 2000) contains alignments and models for 2216 protein families, based on the Swissprot 38 and SP-TrEMBL.

**PRINTS** [http://www.biochem.ucl.ac.uk/bsm/dbbrowser/PRINTS/PRINTS.html](http://www.biochem.ucl.ac.uk/bsm/dbbrowser/PRINTS/PRINTS.html)
PRINTS is a compendium of protein fingerprints.

**INTERPRO** [http://www.ebi.ac.uk/interpro/](http://www.ebi.ac.uk/interpro/)
2990 entries, representing 2373 families, 556 domains, 47 repeats and 14 post-translational modification sites encoded by 4884 different regular expressions, profiles, fingerprints and HMMs. (PFAM, PRINTS, PROSITE)
Database Links (CONT.)

IBM Bioinformatics Grp  http://www.research.ibm.com/bioinformatics
Web access to engines implementing all of the group’s algorithms, plus downloadable Bio-Dictionaries and executable code, description of the group's activities, etc.

EcoCyc  http://ecocyc.panbio.com/ecocyc/
Describes the genome and the biochemical machinery of E. coli. EcoCyc is a literature-derived electronic reference source for E. coli biologists, and for biologists who work with related microorganisms.

Enzyme Database  http://www.expasy.ch/enzyme/
A repository of information relative to the nomenclature of enzymes. It is primarily based on the recommendations of the Nomendature Committee of the International Union of Biochemistry and Molecular Biology (IUBMB) and describes each type of characterized enzyme for which an EC (Enzyme Commission) number has been provided / 15-Jun-2000 (3705 entries)

GPCRDB  http://www.gpcr.org/7tm/ (http://swift.embl-heidelberg.de/7tm/)
Information system for G protein-coupled receptors (GPCRs)

Sequenced genome data and software.

AceDB  http://www.sanger.ac.uk/Software/Acedb/
Acedb is a genome database system that provides a custom database kernel, with a non-standard data model designed specifically for handling scientific data flexibly, and a graphical user interface with many specific displays and tools for genomic data.

WIT  http://wit.mcs.anl.gov/WIT2/
A www-based system to support the curation of function assignments made to genes and the development of metabolic models

A database of single nucleotide polymorphisms
Database Links (CONT.)

DALI http://www2.ebi.ac.uk/dali/
A network service for comparing protein structures in 3D. You submit the coordinates of a query protein structure and Dali compares them against those in the Protein Data Bank.

Protein structure neighbors in Entrez are determined by direct comparison of 3-dimensional protein structures with the VAST algorithm.

FSSP http://www2.ebi.ac.uk/dali/fssp/fssp.html
If you want to know the structural neighbours of a protein already in the Protein Data Bank, you can find them in the FSSP database.

Scop http://scop.mrc-lmb.cam.ac.uk/scop/
Structural Classification of Proteins. 1.50 release / 10650 PDB Entries (29 Feb 2000)

FlyBase http://flybase.bio.indiana.edu:82/
Everything about Drosophila melanogaster

EMP (registration) http://wit.mcs.anl.gov/EMP/
Enzymes and metabolic pathways

JPRED http://jura.ebi.ac.uk:8888/
Consensus-based 2ndary structure prediction

E-motif http://motif.stanford.edu/emotif/
Pattern discovery from *aligned* sequences

The PHYlogeny Inference Package is a package of programs for inferring phylogenies (evolutionary trees).
Database Links (CONT.)

FASTA  http://www2.ebi.ac.uk/fasta3/
Web-based interface for FASTA

Web-based interface for Blast and its variants

Smith-Waterman  http://www2.ebi.ac.uk/bic_sw/
Web-based interface for the Smith-Waterman algorithm

MSA  http://www.ibc.wustl.edu/ibc/msa.html
The MSA multiple sequence alignment algorithm

CLUSTAL-W  http://www.ibc.wustl.edu/msa/clustal.html
The CLUSTALW multiple sequence alignment algorithm

RasMol/Chime  http://www.umass.edu/microbio/rasmol/
Molecular visualization freeware

PIMA  http://dot.imgen.bcm.tmc.edu:9331/multi-align/Options/pima.html
Motif-based multiple sequence alignment based on pairwise comparisons

Molecule analysis, editing and display package

BOXSHADE  http://www.ch.embnet.org/software/BOX_form.html
Tool for shading multiple sequence alignments

ESPript 1.9  http://www-pgml.ipbs.fr:8080/cgi-bin/nph-ESPript_exe.cgi
Tool for coloring multiple sequence alignments
Database Links (CONT.)

**GeneMark**  http://genemark.biology.gatech.edu/GeneMark/
HMM-based tool for discovering coding regions in prokaryotic genomes

**GeneFinder**  http://dot.imgen bcm.tmc.edu:9331/gene-finder/gf.html
Splice sites, Protein coding exons and Gene models construction, Promotor and poly-A search

A graphical analysis tool which finds all open reading frames of a selectable minimum size in a user's sequence or in a sequence already in the database.

**GeneQuiz**  http://jura.ebi.ac.uk:8765/ext-genequiz/
Highly automated analysis of biological sequences

**PROWL**  http://prowl.rockefeller.edu/
A resource for protein chemistry and mass spectrometry

**SAMBA**  http://www.irisa.fr/cosi/SAMBA/
SAMBA is a 128 processor array for speeding up the comparison of biological sequences. The hardware implements a parameterized version of the Smith and Waterman algorithm allowing the computation of local or global alignments with or without gap penalty.

**MOTIF**  http://www.motif.genome.ad.jp/
Automated search of motifs from various libraries in a sequence of interest

**CATH**  http://www.biochem.ucl.ac.uk/bsm/cath/
A novel hierarchical classification of protein domain structures, which clusters proteins at four major levels, class(C), architecture(A), topology(T) and homologous superfamily (H)

**MAGPIE**  http://genomes.rockefeller.edu/magpie/
MAGPIE Automated Genome Project Investigation Environment
Database Links (CONT.)

OWL  http://www.biochem.ucl.ac.uk/bsm/dbbrowser/OWL/OWL.html
A non-redundant composite of 4 publicly-available primary sources: SWISS-PROT, PIR (1-3), GenBank (translation) and NRL-3D

A division of GenBank that contains sequence data and other information on "single-pass" cDNA sequences, or Expressed Sequence Tags, from a number of organisms. 4,334,336 entries by June 09, 2000.

DSSP  http://swift.embl-heidelberg.de/dssp/ ???
The DSSP database is a database of secondary structure assignments (and much more) for all protein entries in the Protein Data Bank (PDB)

Grail  http://compbio.ornl.gov/Grail-1.3/
Gene recognition and assembly internet link

PRATT  http://www2.ebi.ac.uk/pratt/
Web server for a pattern discovery algorithm

PHD  http://www.embl-heidelberg.de/predictprotein/predictprotein.html
A service for sequence analysis, and structure prediction.

MGD  http://www.informatics.jax.org/
Mouse genome information

Demo Program for NTI Viewer

Pedro (Coutinho)'s Site  http://www.public.iastate.edu/~pedro/rt_all.html
Links to tools galore
Useful Notation And Definitions From Computer Science

\[ f(n) = \mathcal{O}(g(n)) \]

- \( f(n) \) is said to be \( \mathcal{O}(g(n)) \) -- "big-Oh of \( g(n) \)" iff there exist constants \( c \) in \( \mathbb{R} \) and \( n_0 \) in \( \mathbb{N} \) such that \( f(n) \leq c \times g(n) \) for all \( n \geq n_0 \)

\[ f(n) = \Omega(g(n)) \]

- \( f(n) \) is said to be \( \Omega(g(n)) \) -- "big-Omega of \( g(n) \)" iff there exist constants \( c \) in \( \mathbb{R} \) and \( n_0 \) in \( \mathbb{N} \) such that \( f(n) \geq c \times g(n) \) for all \( n \geq n_0 \)

\[ f(n) = \Theta(g(n)) \]

- iff \( f(n) = \mathcal{O}(g(n)) \) and \( f(n) = \Omega(g(n)) \)
Recurrence Equation

- Defined in terms of ... itself

\[ T(n) = f(T(1), T(2), T(3),...,T(n-1), n) \]

Examples:

\[ T(1) = 1 \]
\[ T(n) = 2T(n-1) + 1 \]

\[ T(1) = 1 \]
\[ T(n) = T(n-1) + n \]

\[ T(1) = 1/3 \]
\[ T(n) = T(n-1) + 1/(2n-1)/(2n+1) \]
Recurrence Equation (cont.)

- The concept of "recursion"
  
  \[
  \text{factorial}(0) = 1 \\
  \text{factorial}(n) = \text{factorial}(n-1) \times n
  \]

```c
int
factorial(int n)
{
    if ( n == 0 ) {
        return(1) ;
    }
    else {
        return ( n * factorial(n-1) ) ;
    }
}
```

Q: is there anything wrong with this piece of code?
Sorting Numbers

- **Input:** a set of $N$ many real numbers
  - **Output:** the same set in order of increasing value

Theorem: "any algorithm that sorts $n$ numbers by comparisons requires $\Omega(n \log n)$ comparisons"

- Algorithms: $\text{BubbleSort}(S,N)$
  - $\text{* *QuickSort}(S,N)$

- Running times?

- "Efficient Algorithm": running time is a polynomial function of the input size $N$
Sorting Numbers (cont.)

QuickSort(S,N)
{
    - if N=1 then return S

    - pick random element r in S
    - separate S into sets
        S_1 of elements that are < r,
        S_2 of elements = r, and S_3 of elements > r
    - return the result of
        ( QuickSort(S_1, |S_1|), S_2, QuickSort(S_3, |S_3|) )
}
"Efficient algorithm" running time is $O(p(n))$

not "Efficient algorithm"

NP-complete problems

NP-hard problems: all optimization problems whose decision versions are NP-complete.
More Definitions

Graph
A set V of points (=vertices) and a set E of lines (=edges) that connect pairs of points.
Notation: G = (V,E) with V={v₁, v₂, ..vₙ} and E={ (u,v) / u, v in V }

undirected graph: all edges are UNordered

directed graph: all edges (u,v) are ORdered
u is the tail and v is the head

incident vertex/edge: an edge (u,v) is incident on the vertices u and v -- vertices u and v are incident on edge (u,v)

degree of a vertex: the number of vertices that are adjacent to it (in-degree & out-degree in directed graphs)
More Definitions (cont.)

weighted: each edge is associated with a real number known as 'distance', 'weight' or 'cost'

path: an ordered list \((v_1, v_2, \ldots v_k)\) of vertices such that \((v_i, v_{i+1})\) is an edge of the graph

cycle in an UG: a path such that \(v_1 = v_k\) and no edge is repeated

cycle in a DG: a path such that \(k > 1\) and \(v_1 = v_k\)

subgraph of a graph: a graph \(G'=(V',E')\) where \(V' \subseteq V\) and \(E' \subseteq E\)

acyclic: a graph without cycles
More Definitions (cont.)

complete UG: a graph where for every pair $u, v$ in $V$ the edge $(u,v)$ is in $E$

complete DG: a graph where for every pair $u, v$ in $V$ the edges $(u,v)$ and $(v,u)$ are in $E$

bipartite: a graph whose vertices can be separated into two sets $V_1$ and $V_2$ such that for every edge $(u,v)$ in $E$ we have $u$ in $V_1$ and $v$ in $V_2$

connected UG: iff every vertex of the graph can be reached from every other vertex of the graph

strongly connected DG: iff every vertex can be reached from every vertex
More Definitions (cont.)

weakly connected DG: / disregard directions

not-connected DG: / if neither strongly nor weakly

interval graph (undirected): begin with a collection of intervals on the real line; create a vertex for each interval in the collection to build V; for any two intervals u and v with non-zero intersection enter (u,v) in E

adjacency matrix of a graph: a $|V| \times |V|$ matrix $M$ with $M(i,j) = 1$ if $(v_i, v_j)$ is in E, 0 otherwise. If the graph is weighted $M(i,j)$ is the weight of the respective edge
More Definitions (cont.)

Tree: A directed acyclic graph that a) has a root vertex that no edges enter, b) every vertex other than the root has one edge entering it, and c) there is a unique path from the root to every vertex.

leaf: a vertex with no outgoing edges

parent & child: if (u,v) is in E then u (resp. v) is the parent (resp. child) of v (resp. u)

depth of a vertex: the length of the path from the root to the vertex

height of a vertex: the length of the longest path from the vertex to any leaf

least common ancestor:
More Definitions (cont.)

Undirected (rooted) Tree: A connected, undirected acyclic graph with one vertex distinguished as the root.

Graph Traversal: Traverse (visit) all of the vertices of the graph.

'Popular' Traversal Schemes:
Depth-First
Breadth-First
Spanning Tree

Definition: Let \( G=(V,E) \) be an undirected, connected graph. Its spanning tree is an undirected tree \( S=(V,T) \) and its cost is the sum of the weights of the edges in \( T \).

**Minimum Cost** Spanning Tree

The Minimum Cost Spanning Tree Property

Prim's algorithm: Make the greedy choice

Running time: \( O(n^2) \)
Tree Traversals

Popular traversals: DFS and BFS

Depth First Traversal

dfs(v) {
  mark v as "visited";
  for each vertex w that is adjacent to v
    if ( w has not been "visited" )
      dfs(w);
  end-if
end-for
}

Breadth First Traversal

L <- {v};

bfs(L) {
  if ( L is not empty )
    let f be the first element of L;
    mark f as "visited";
    remove f from L;
    for each vertex w that is adjacent to f
      if ( w has not been "visited" )
        mark w as "visited";
        append w to L;
      end-if
    end-for
  end-if
}
Tree Traversals - Example

DFS

BFS
Simple And Otherwise

Simple Problems
- Given a DG/UDG, find a cycle that includes every edge of the graph only once (Eulerian graph)
- Given an UDG that is connected find a minimum spanning tree for it
- Given an UDG find a maximum cardinality subset of the edges such that no two edges share a vertex

Seemingly Simple Problems
- Given a DG/UDG, find a cycle that includes every vertex of the graph only once - except for first/last vertex (Hamiltonian graph)
- Given an UDG with cost on each edge, find a Hamiltonian cycle of minimum cost (TSP)
- Given an UDG find the minimum number of colors needed to color it so that no two adjacent vertices have the same color
Maps

- Genetic Linkage map
  - 10-100M bp
  - order and relative distance among genes

- Physical Map
  - 0.1-1M bp
  - maps showing actual distance among genes

- Sequencing
  - 1-10K bp
  - actual contents of genes
Maps (cont.)

- Full DNA

  - cut and clone into overlapping YAC clones: 0.1-1 Mbp

  - cut and clone into overlapping cosmid clones: 10-50 Kbp

  - sequence by shotgun: 1 Kbp
Maxam-Gilbert Sequencing Method

DNA-sequencing method developed by Maxam and Gilbert. This method uses chemical reagents to destroy specific nucleotide bases and thus break the DNA molecule at specific sites. First the strands of the DNA molecule are labeled radioactively at one end (usually the 5' end), and the two strands are separated (only one will be sequenced). Then aliquots of the chosen strand are treated with four different chemical reagents that break the strand at one or two specific nucleotides; the treatment is limited so that at most a single residue of the susceptible base(s) in the molecule will react. Thus, in each reaction mixture, a nested set of radioactive fragments is generated, as shown here for only the reaction mixture that destroys C residues. Finally, gel electrophoresis is used to separate the products of each reaction by size. The pattern of radioactive bands seen on X-ray film immediately reveals the sequence.

Sanger-Coulson Sequencing Method

DNA-sequencing method developed by Sanger. A dideoxynucleotide is incorporated into a growing DNA strand, subsequently stopping chain growth, since it cannot form a phosphodiester bond with the next incoming nucleotide. Four different reactions are run, each with a different dideoxynucleotide. The products of each reaction are a series of incompletely elongated segments, which are separated by gel electrophoresis. As in the Maxam-Gilbert method, the sequence can be read from the bands produced in the gel.


Fluorescence

Figure from: http://dna.ctandct.com/PGG/PGG.html
Physical Maps

- fingerprints
How To Get Fingerprints

- Restriction Site Mapping
  - fingerprint is the length of the fragment
  - double digest
    - A
    - B
    - A+B
  - partial digest

- Errors:
  - inaccuracy in measuring lengths
  - fragments too short
How To Get Fingerprints (cont.)

- Hybridization Mapping
  - probes & clones
  - fingerprint is the set of probes hybridizing to the clone

- Errors:
  - false positives
  - false negatives
  - chimeric clones
interval graphs
From Hybridization To Maps

- Assumptions:
  - probes are "unique"
  - there are no errors
  - all "clones x probes" have been carried out

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>C2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>C3</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>C4</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>C5</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C7</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
From Hybridization To Maps (cont.)

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>C3</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>C5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>C6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>C7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>C8</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
From Hybridization To Maps (cont.)

- pick any two rows $i, j$
- let $S_i$ be the set of columns where there is 1's in row $i$

- cases:
  - $S_i$ intersection $S_j = \text{empty}$
  - $S_i$ subset of $S_j$ (or vice versa)
  - non-zero intersection
From Hybridization To Maps (cont.)

let's work with this specific example:

<table>
<thead>
<tr>
<th></th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>C2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>C3</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

\{2,7,8\}  \{2,7,8\}  \{2,7,8\}
From Hybridization To Maps (cont.)
From Hybridization To Maps (cont.)

back to the original example:

<p>| | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>C2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>C3</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>...</td>
</tr>
<tr>
<td>C4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>C6</td>
<td></td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>...</td>
</tr>
<tr>
<td>C7</td>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>...</td>
</tr>
<tr>
<td>C8</td>
<td></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>...</td>
</tr>
</tbody>
</table>
From Hybridization To Maps (cont.)

- Summing it up:
  - build a uDG with one vertex for each row
  - an edge connects two vertices if the respective $S_i$'s intersect (only!)

  FIX columns within a group
  - traverse graph in DF order
  - at every vertex apply Place($u$, $v$, $w$) (see Setubal/Meidanis Ch. 5)

- Join groups
  - build a DG with an edge between groups A and B iff all of B's rows are subsumed by some row in A
  - process vertices in topological order gluing components together
Fragment Assembly

- We want to sequence entire molecule directly
- ... but we cannot / we can only have small-size fragments

- Start with "shotgun" and generate large number of fragments in 200-700 bp range
- Use estimated size of target and overlap information as guide
- Report "consensus" sequence
Fragment Assembly (cont.)

- Problems:
  - Errors
    - base calling / chimeric frags. / contamination
  - Unknown orientation
  - Repeated regions
    - AXBXCXD, AXBYCXDYE, X...X
  - No coverage
Fragment Assembly (cont.)

- **SCS:**
  Definition: Given a collection \( F \) of strings, find the shortest possible string \( S \) s.t. for every \( f \) in \( F \), \( S \) is a superstring of \( f \).
  - does not allow for experimental errors
  - orientation must be known
  - if repeats present, answer will be *shorter*

- **Reconstruction**
  Definition: Given a collection \( F \) of strings and a tolerance \( e \) in \([0, 1]\) find the shortest possible string \( S \) s.t. for every \( f \) in \( F \)
  \[
  \min(d(f,S), d(f',S)) \leq e \cdot |f|
  \]
  - can cope with errors and orientation
  - cannot handle chimeric frags. / repeats / lack of coverage
Fragment Assembly (cont.)

- MultiContig
  Definition: Given a collection $F$ of strings, an integer $t$ and a tolerance $e$ in $[0,1]$ find the minimum number of sub-collections $C_i$ $1 \leq i \leq k$ s.t. every $C_i$ admits a $t$-contig with $e$-consensus
  - models errors, orientation and gaps
  - cannot use size of target
  - partial success with repeats