Practice E-R Diagram Problems from prior midterm exams

Mutations, signaling pathways, and disease. There is much interest in prioritizing genetic mutations, discovered through sequencing experiments, based on their likely affect on human disease. One way that mutations can affect disease is through disruption of protein-protein interactions in signaling pathways. In these pathways, which transmit information from one part of a cell to another, proteins form complexes through interaction of their three-dimensional shapes. If a mutation disrupts a binding site on a protein, it can affect the ability of that protein to successfully bind to its interaction partners. This can then disrupt the ability of the signals to be successfully transmitted and may lead to disease. The goal of this question is to design a database which can be used for storing information to help prioritize mutational affects.

A. Design and draw an **ER diagram** that captures the information below. Create entities and relations corresponding to the **bold face names** in the statements. Be sure to indicate:

- **the attributes for each entity (**and relation attributes if any**),**
- **keys for each entity and relation** (underline key fields),
- **relationship class** (one-to-many, many-to-many, etc.),
- **participation constraints** (total, partial),
- any **constraints that cannot be captured in the ER diagram**.

Use the symbols on the reference page in your diagram.

1. Each signaling **pathway** has an id, a name, and a function.
2. Each **protein** has an id, a name, a sequence, and a three-dimensional structure (stored as an external code from the protein data bank – PDB).
3. Proteins have pairwise **interact**ions. Each protein can have zero or more interactions. No more than one interaction can involve the same two proteins.
4. Each protein-protein interaction **participates in** exactly one signaling pathway or in none. Each pathway has at least one interaction.
5. A single nucleotide variation or **SNV**, has an id, a chromosome number and nucleotide position (where it occurs), a reference nucleotide, and a variant nucleotide.
6. A **binding site** has an id and a three-dimensional shape description (text).
7. Each protein **contains** zero or more binding sites. Each binding site is contained in exactly one protein.
8. An SNV can **alter** zero or more binding sites. Each binding site can be altered by one or more SNVs. When an SNV alters a binding site, a magnitude value is included to quantify the degree of alteration.
9. A **disease** has an id and a name.
10. A signaling pathway may be **involved** with zero or more diseases. Each disease is involved with at least one pathway.

Gene function annotation through inheritance. A common task of functional genomics is to predict cellular function(s) for a newly discovered gene. One method involves translating a gene into its protein sequence, assigning the sequence to one or more protein families based on sequence similarity, and then transferring functional annotations from the family to the gene. The goal of this question is to design a database which can be used for this type of annotation by inheritance of function.

A. Design and draw an **ER diagram** that captures the information below. Create entities and relations corresponding to the **bold face names** in the statements. Be sure to indicate:

- **the attributes for each entity (**and relation attributes if any**),**
- **keys for each entity and relation** (underline key fields),
- **relationship class** (one-to-many, many-to-many, etc.),
- **participation constraints** (total, partial),
- any **constraints that cannot be captured in the ER diagram**.

Use the symbols on the reference page in your diagram.

11. Each **gene** has an id, an abbreviation, and a common name.
12. Each **genome** has an id, a scientific name, and a common name.
13. A **build** has an id and a short name. (A build is an assembly of a genome. New builds are produced as the genomic sequence is refined and corrected. For example, recent human genome builds are called hg19, hg38. This is explanatory and not part of the question.)
14. Each genome has one or more **version**s, that is, one or more builds. Every build has exactly one genome.
15. Every gene has **genetic coordinates** in one or more builds. Coordinates consist of a chromosome, a start location, and an end location. Each build has genetic coordinates for zero or more genes.
16. A **protein** has an id and a sequence.
17. Each gene has one or more protein **translation**s (corresponding to alternative splicings). Every protein has exactly one gene for which it is the translation.
18. A **PFAM** is a protein family. It has an id, a name, and a Hidden Markov Model (HMM) describing the protein family sequence.
19. Every PFAM has one or more **member** proteins. Every protein is a member of zero or more PFAM families.
20. A **function** has an id and a name. (Examples of function are specific, but could fall into broad categories such as enzymatic reactions, hormone receptors, molecular transport, DNA binding. This is explanatory and not part of the question.)
21. Every protein family **performs** zero or more functions. Every function is performed by zero or more protein families.

MicroRNA and mRNA expression covariance. You are building a database to identify microRNAs and mRNAs that appear to interact in different tissue types based on differential expression analysis. MicroRNAs (miRs) are small non-coding RNAs that regulate the expression of target mRNAs through complementary binding which initiates the action of a protein complex resulting in either 1) inhibition of translation of the mRNA to protein or 2) degradation of the mRNA prior to translation. In the latter case, an increase in miR expression should result in a decrease in target mRNA expression and vice-versa. Your database will help identify microRNA-mRNA pairs where reciprocal expression co-variance occurs. It will store experiments in which single miRs are overexpressed and which record the expression of the mRNAs.

A. Design and draw an **ER diagram** that captures the information below. Create entities and relations corresponding to the **bold face names** in the statements. Be sure to indicate:

- **the attributes for each entity (**and relation attributes if any**),**
- **keys for each entity and relation** (underline key fields),
- **relationship classification** (one-to-many, many-to-many, etc.),
- **participation classification** (total, partial),
- any **constraints that cannot be captured in the ER diagram**.

Use the symbols on the reference page in your diagram.

1. A **miR** has an id, a name, a species, and a sequence.
2. A **gene** has an id, a name, a species, and a sequence.
3. An **experiment** has an id, a date, and a description of the experimental conditions.
4. A **sample** has an id, a species, and a tissue type.
5. A sample is **use**d one or more experiments. Each experiment uses exactly one sample.
6. A miR may be **overexpressed** in an experiment. The level of overexpression (over_exp, a floating point number) is recorded. A miR may be overexpressed in zero or more experiments. Each experiment involves overexpression of zero or one miRs. (In the case of zero, the experiment is a control, this is for your understanding, it does not have to be recorded).
7. An experiment determines the **expression_level** of a gene. The level of expression (exp_level, a floating point number) is recorded. Each experiment determines the expression level of many genes and the expression level of a gene is determined by zero or more experiments.
8. A gene is a putatitve **target** of a miR if the core sequence of the miR is complementary to some part of the gene sequence. Each gene can be the putative target of zero or more miRs and each miR has zero or more genes which are putative targets.

Taxonomically restricted genes. You are building a database to identify new genes that have arisen through evolution. Such genes are important for the origin of new species and the emergence of new traits, for example, morphological (body) differences, metabolic differences, pathogenicity, and environmental adaptation. We will assume that each new gene arises once in evolutionary history in an ancestral genome and is passed down as an inheritance to one or more descendant, present-day genomes. Thus, when considering the phylogenetic tree of life, a new gene will appear only in a single subtree containing the descendants of the origin genome. The root of the subtree determines the relative age of origin and species distribution of the new gene. Your database will be built on gene sequence data and taxonomy data. BLAST searches will be used to determine which genes from one organism are related to genes in another organism.

A. Design and draw an **ER diagram** that captures the information below. This can be a disjoint ER diagram where you draw the same entity tables multiple times for different relationships. Create entities and relations corresponding to the **bold face names** in the statements. Be sure to indicate:

- **the attributes for each entity (**and relation attributes if any**),**
- **keys for each entity and relation** (underline key fields),
- **relationship classification** (one-to-many, many-to-many, etc.),
- **participation classification** (total, partial),
- any **constraints that cannot be captured in the ER diagram**.

Use the ER Diagram Legend symbols on the reference page in your diagram.

1. A **taxon** is a node in a phylogenetic tree. The tree is implicit and not a database table. Each taxon has an id, a rank, a scientific name, and may have a common name. A rank is one of the following strings: (‘species’, ‘genus’, ‘family’, ‘order’, ‘class’, ‘phylum’, ‘kingdom’). Examples of taxa are 1) *Escherichia col*, with rank ‘species,’ and 2) *Hominidae* with rank ‘family.’ (Taxa is the plural of taxon.)

The implicit phylogenetic tree arises from the following relationship.

2. Each taxon, except the topmost in the phylogenetic tree has a **parent** which is another taxon. Not all taxa are parents. One taxon may be parent to multiple taxa.
3. Each **gene** has an id and a DNA sequence.
4. Each gene has exactly one **source_genome** which is the taxon of the species that contains the gene. A taxon may be the source_genome for zero or more genes.
5. A gene (query gene) has **BLAST hits** with zero or more other genes (hit genes). Not all genes are hit genes. Some genes are hit genes to more than one query gene. A BLAST hit has an associated e-value and an alignment which is a string.
6. The **highest_node** for a gene is the lowest taxon in the phylogenetic tree for which all the BLAST hits for the gene are below that node. This is calculated after BLAST hits are calculated. A gene will have a highest node only if it has BLAST hits. A taxon may or may not be a highest node and it may be a highest node for more than one gene.

Function enrichment of differentially expressed genes. You are building a database to test gene set enrichment following experiments that measure differential expression of genes. You want to determine if the differentially expressed genes belong to any of a collection of pre-defined genes sets at a higher rate than would be expected by chance. This would suggest that the particular function(s) of the gene sets are being impacted by the experimental conditions and being mediated through the genes that are differentially expressed. The pre-defined gene sets are typically curated by experts in the field of gene function, or from literature searches. You will also include part of the Gene Ontology hierarchy to list functions of individual genes. Your database will be built on a small set of well studied species, but will apply more broadly through the use of orthologs to match a gene from one species to its most similar gene in another species.

A. Design and draw an **ER diagram** that captures the information below. Create entities and relations corresponding to the **bold face names** in the statements. Be sure to indicate:

- **the attributes for each entity (**and relation attributes if any**),**
- **keys for each entity and relation** (underline key fields),
- **relationship classification** (one-to-many, many-to-many, etc.),
- **participation classification** (total, partial),
- any **constraints that cannot be captured in the ER diagram**.

Use the symbols on the reference page in your diagram.

7. A **gene** has an id, a name, and a species name (genus and species).
8. **Ortholog** is a strong similarity relationship between a gene in one species and a gene in another. Each gene has zero, one, or more orthologs.
9. A pre-defined **gene set** has an id, a name, a count of genes, and a description.
10. Genes are **members** of a pre-defined gene set. Each pre-defined gene set has at least one member. Each gene can be a member of zero, one or more pre-defined gene sets.
11. A **differential set** has an id, a count of genes, and a description.
12. A differential set **includes** genes. Each differential set includes at least one gene. Each gene can be included in zero, one, or more differential sets.
13. **Users** of your database have an id (no name will be required because usage will be anonymous).
14. A user **submits** a differential set. Each user submits at least one differential set and each differential set is submitted by exactly one user.
15. A **GO term** has an id and a description.
16. Each GO term is part of the **GOTree** which is a parent and child relationship. Each term is the parent of zero, one, or more terms and each term is the child of zero or one term.
17. Each gene has a **function** described by zero or one GO term. Each GO term can be the function of zero, one, or more genes.

Reference Page

Create Table Format

```
CREATE TABLE tbl_name (
    col_name data_type [NOT NULL | NULL][DEFAULT default_value] [AUTO_INCREMENT],
    PRIMARY KEY (index_col_name,...),
    FOREIGN KEY (index_col_name,...) REFERENCES tbl_name (index_col_name,...)
  [ON DELETE {CASCADE | SET NULL | NO ACTION | SET DEFAULT}]
  [ON UPDATE {CASCADE | SET NULL | NO ACTION | SET DEFAULT}]
) ENGINE = INNODB
```

data type:

INT[(length)]
REAL[(length,decimals)]
DOUBLE[(length,decimals)] [UNSIGNED] [ZEROFILL]
FLOAT[(length,decimals)] [UNSIGNED] [ZEROFILL]
DECIMAL(length,decimals) [UNSIGNED] [ZEROFILL]
CHAR(length) [BINARY | ASCII | UNICODE]
VARCHAR(length) [BINARY]
DATE
TIME
TEXT
ENUM(value1,value2,value3,...)

Select Statement Format

SELECT select_expression
[FROM table_references]
[WHERE where_definition]
[GROUP BY col_list]
[HAVING having_definition]
[ORDER BY col_list [ASC | DESC]]
[LIMIT row_count]

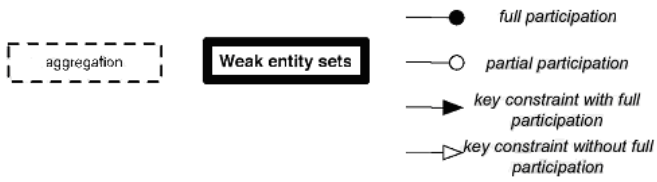
Using LIKE:
LIKE matches the entire string
% means zero or more characters
_ (underscore) means any one character

examples (TRUE if):
LIKE '%pat%' – pat somewhere in string
LIKE 'pat%' – pat at beginning of string
LIKE '_pat%' – pat starting at second character

Using REGEXP:
REGEXP matches anywhere in the string

example (TRUE if):
REGEXP 'pat1|pat2|pat3' – pat1 or pat2 or pat3 anywhere in string

ER Diagram Legend



Create Index Format

```
ALTER TABLE table_name
ADD PRIMARY KEY (field_name,…)

ALTER TABLE table_name
ADD [UNIQUE|INDEX] index_name (field_name,…)
```