-- B. Crump, A. Jarrell

**Functional Dependencies and 3NF**

Person:

per\_id —> name, address+, email, gender, zip\_code

(left side is a primary key/ candidate key)

per\_id —>—> phone (multi-valued attribute)

address+ —> zip\_code (each address is associated with a zip\_code)

name, address+, email, gender, zip\_code —> per\_id

(left side can also be a candidate key)

Company:

comp\_id —> address+, zip\_code, website, primary\_sector

(left side is a primary key/ candidate key)

address+ —> zip\_code (each address is associated with a zip\_code)

comp\_id —>—> specialty (multi-valued attribute)

address+, zip\_code, website, primary\_sector —> comp\_id

(left side can also be a candidate key)

Course:

c\_code —> title, level, description, status, retail\_price

(left side is a primary key/ candidate key)

title —> description (each title is associated with a description)

Decompose into:

course1(c\_code, title, level status, retail\_price)

course2(title, description)

since description is not part of a key in course. When we intersect course1 and course2, we get title, which is the superkey of course2.

Section:

c\_code, sec\_no, semester, year —> offered\_by, format, price, complete\_date

(left side is a primary key/ candidate key)

c\_code, sec\_no, semester, year, offered\_by, format —> price

(left side is associated with a price)

Job\_Profile:

pos\_code —> title, description, avg\_pay

(left side is a primary key/ candidate key)

pos\_code —>—> required\_skill (multi-valued attribute)

title —> description (each title is associated with a description)

Decompose into:

jp1(pos\_code, title, avg\_pay)

jp2(title, description)

since description is not part of a key in job\_profile. When we intersect jp1 and jp2, we get title, which is the superkey of jp2.

Job:

job\_code —> type, pay\_rate, pay\_type, company, pos\_code

(left side is a primary key/ candidate key)

Knowledge\_Skills:

ks\_code —> title, description, level

(left side is a primary key/ candidate key)

title —> description (each title is associated with a description)

Decompose into:

ks1(ks\_code, title, level)

ks2(title, description)

since description is not part of a key in knowledge\_skills. When we intersect ks1 and ks2, we get title, which is the superkey of ks2.

Job\_Company:

job\_code —> comp\_id

job\_code, comp\_id —> (all attributes associated with each)

Teaches:

ks\_code, c\_code —> (all attributes associated with each)

Has\_Job:

job\_code —> per\_id

per\_id, job\_code —> (all attributes associated with each), start\_date, end\_date

Skills:

ks\_code, pos\_code —> (all attributes associated with each)

Experience:

per\_id, ks\_code —> (all attributes associated with each)

Takes:

per\_id, c\_code, sec\_no, semester, year —> (all attributes associated with each)

Offers:

comp\_id, c\_code, sec\_no, semester, year —> (all attributes associated with each)