Program Patterns:
Reorganization & Derivation Patterns

J.W. Choi

2024

Data Processing Program Patterns

- Data Search
- Data Update
- Data Copying & Moving
- Data Transformation
- Data Reorganization
- Data Derivation

Roadmap

- Data reorganization
- Data derivation

Data Reorganization

- sorting
- grouping
- ordering
- sampling
- decomposition
  - vertical
  - horizontal
- merging

Data Grouping

- Group data for easy visualization and efficient statistics computation
- Types of grouping
  - grouping on one attribute (struct member)
  - grouping on multiple attributes
  - grouping without order
  - grouping with order

Example: Grouping with Ordering

Name	Age	Hobby
Kim	20	swim
Lee	45	music
Youn	44	poker
Choi	20	swim
Han	30	poker
Ko	30	movie

by Age

Name	Age	Hobby
Kim	20	swim
Choi	20	swim
Han	30	poker
Ko	30	movie
Youn	44	poker
Lee	45	music

by Hobby

Name	Age	Hobby
Ko	30	movie
Lee	45	music
Kim	20	swim
Choi	20	swim
Youn	44	poker
Han	30	poker

Lab: Grouping (with No Ordering)

- Write the following C program:
  - Read data from file personal.txt
    - Each data has name, age, and hobby
    - Assumption: name and hobby are both SINGLE WORDs
  - Group data by age and write to file age.txt
  - Group data by hobby and write to file hobby.txt

Sampling

- When there is a large number of data, take only a small sample of the data for fast statistics computation.
- Need to determine the acceptable sample size.
- Sampling methods
  - random sampling
  - n-th name selection (take every n-th data)
  - stratified sampling (group first, then do random sampling within each group)
  - …

Example: n-th name selection

Name	Age	Hobby
Kim	20	swim
Lee	45	music
Youn	44	poker
Choi	20	swim
Han	30	poker
Ko	30	movie

→

Name	Age	Hobby
Kim	20	swim
Youn	44	poker
Han	30	poker

sampling odd numbered rows

Lab

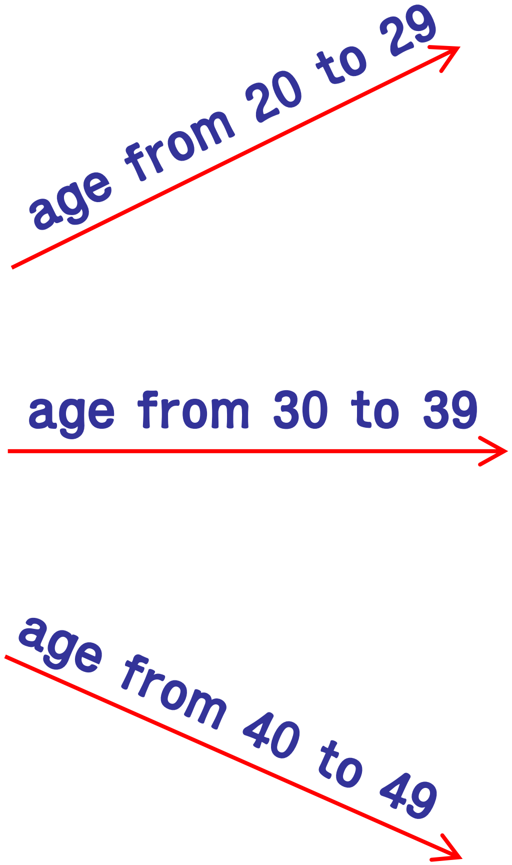
- Write the following C program:
  - Read data from file vote.txt
    - Each line contains (region, age, candidate voted)
  - Compute percentage of votes each candidate received and print to terminal
    - e.g., Washington 40.7%, Lincoln 30.2%, Clinton 29.1%
  - Sample every third row
    - Compute and print to terminal votes for each candidate
  - Sample every tenth row
    - Compute and print to terminal votes for each candidate

Data Decomposition

- Horizontal decomposition
- Vertical decomposition

Horizontal Decomposition: Example

Name	Age	Hobby
Kim	20	swim
Lee	45	music
Youn	44	poker
Choi	20	swim
Han	30	poker
Ko	30	movie



age from 20 to 29

Name	Age	Hobby
Kim	20	swim
Choi	20	swim

age from 30 to 39

Name	Age	Hobby
Han	30	poker
Ko	30	movie

age from 40 to 49

Name	Age	Hobby
Lee	45	music
Youn	44	poker

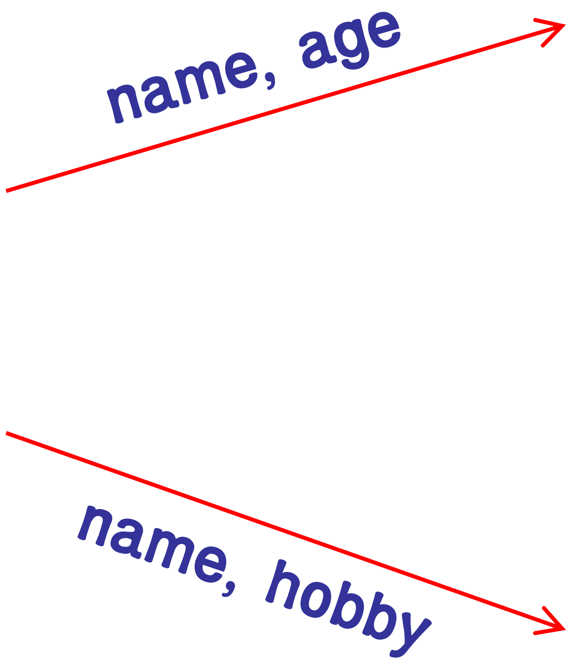
Lab

- Write the following C program:
  - Read data from file personal.txt
    - Each line contains name, age and hobby
    - Store data in a struct array
  - Decompose data by age as follows
    - age from 10 to 19
    - age from 20 to 29
    - age from 30 to 39
    - age from 40 to 49
  - Generate a struct array for each age range and write to file output.txt in the following format

```
Age from 10 to 19
--------------------

Age from 20 to 29
--------------------
Kim    20    swim
Choi   20    swim

Age from 30 to 39
--------------------
Han    30    poker
Ko     30    movie

Age from 40 to 49
--------------------
Lee    45    music
Youn   44    poker
```

Vertical Decomposition: Example

Name	Age	Hobby
Kim	20	swim
Lee	45	music
Youn	44	poker
Choi	20	swim
Han	30	poker
Ko	30	movie



Name	Age
Kim	20
Lee	45
Youn	44
Choi	20
Han	30
Ko	30

Name	Hobby
Kim	swim
Lee	music
Youn	poker
Choi	swim
Han	poker
Ko	movie

Lab

- Write the following C program:
  - Read data from file personal.txt
    - Each line contains name, age and hobby
    - Store data in a struct array
  - Generate a struct array that has only name and age, and write to file age.txt
  - Generate a struct array that has only name and hobby, and write to file hobby.txt

Roadmap

- Data reorganization
- Data derivation

Data Derivation

- data aggregation
  - sum, total, average, max, min
- data versioning
- data lineage (data catalog)

Versioning: Example

- Version number tells which data is the newest

salary ver. 1

Name	Age	Salary
Kim	20	50000.0
Lee	45	100000.0
Youn	44	90000.0
Choi	20	45000.0
Han	30	75000.0
Ko	30	70000.0

salary ver. 2

Name	Age	Salary
Kim	20	50000.0
Lee	45	110000.0
Youn	44	99000.0
Choi	20	45000.0
Han	30	75000.0
Ko	30	70000.0

salary ver. 3

Name	Age	Salary
Kim	20	50000.0
Lee	45	110000.0
Youn	44	99000.0
Choi	20	45000.0
Han	30	90000.0
Ko	30	84000.0

10% raise for employees aged from 40 to 49

20% raise for employees aged from 30 to 39

Versioning: Lab Part 1

- Write the following C program:
  - Read data from file salary_v1.txt
    - Each line contains name, age and salary
    - Store data in a struct array
  - Update data so that salaries of employees aged from 40 to 49 are raised 10%
  - Write to file salary_v2.txt
  - Read data from file salary_v2.txt
  - Update data so that salaries of employees aged from 30 to 39 are raised 20%
  - Write to file salary_v3.txt

Versioning: Lab Part 2

- Write the following C program:
  - Compare salary_v1.txt and salary_v3.txt
  - Write to screen the difference between two versions as follows:

```
Kim   20   50000.0 →  50000.0
Lee   45  100000.0 → 110000.0
Youn  44   90000.0 →  99000.0
Choi  20   45000.0 →  45000.0
Han   30   75000.0 →  90000.0
Ko    30   70000.0 →  84000.0
```

End of Class