Practice #3

Performance Measurement ~ String

Yunmin Go

School of CSEE

HANDONG GLOBAL UNIVERSITY

Practice #3 TO-DO List

To-Do	Submission	Notes
Performance Measurement	Screenshot and source code (arrayperf.c)	p.97 ~ p98, Chapter 1 p.22, Chapter 2
Contacts	Screenshot and source code (contacts.c)	p.36, p.46, Chapter 2
String Insertion	Screenshot and source code (strnins.c)	p.47 ~ p.48, Chapter 2

- Upload your screenshot and source codes on LMS by 11pm on 3/16 (Tue).
  - All your screenshots should be merged in one pdf file, screenshot.pdf.
  - Your pdf and all source codes should be compressed into zip file.
  - Your zip file contains four files, screenshot.pdf, arrayperf.c, contacts.c, and strnins.c.
- File name: practice03_Your Student ID_Name.zip (only zip, not pdf, docx, c, etc)
  - ex) practice03_20400022_고윤민.zip

Performance Measurement

- Compare the performance of row-major format and that of column-major format by measuring the execution time
  - Skeleton code: arrayperf.c
  - Refer to p.97-98, Chapter 1 and p.22, Chapter 2.
  - Use clock() to measure the performance.
    - Example

You can also use time() or gettimeofday()

```
#include <time.h>
int main()
{
    clock_t start = 0, stop = 0;
    double duration = 0;
    start = clock();

    // routines to measure execution time

    stop = clock();
    duration = ((double) (stop – start) / CLOCKS_PER_SEC;
    return;
}
```

Performance Measurement

- Expected results

```
PS C:\ds\practice03\sol> .\arrayperf.exe
Time for row-major format: 0.272000
Time for column-major format: 0.617000
PS C:\ds\practice03\sol> .\arrayperf.exe
Time for row-major format: 0.278000
Time for column-major format: 0.616000
PS C:\ds\practice03\sol> .\arrayperf.exe
Time for row-major format: 0.296000
Time for column-major format: 0.685000
```

Contacts with Structure

- Read and parse contact data from a file (data.dat)
  - Skeleton code: contacts.c
  - Refer to p.36, p.46, Chapter 2.
  - Each line of file (data.dat) is in the following format.
    - name; birthday; email; phone_number → Strings
      - ex) yunmin go; 20210302; yunmin@handong.edu; 010-1234-5678
  - Read the line and parse each value by ";".
    - Use strtok() for parsing
  - Store the values into array of structure named 'contacts'

data.dat

```
Iron Man;20210301;tony@avengers.com;111-2222-3333;
Spider Man;20210401;peter@avengers.com;444-5555-6666;
Captain America;20210501;steve@avengers.com;777-8888-9999;
Black Widow;20210601;natasha@avengers.com;123-4567-8901;
Thor;20210701;thor@avengers.com;333-5555-7777;
Hulk;20210801;bruce@avengers.com;777-8888-9999;
```

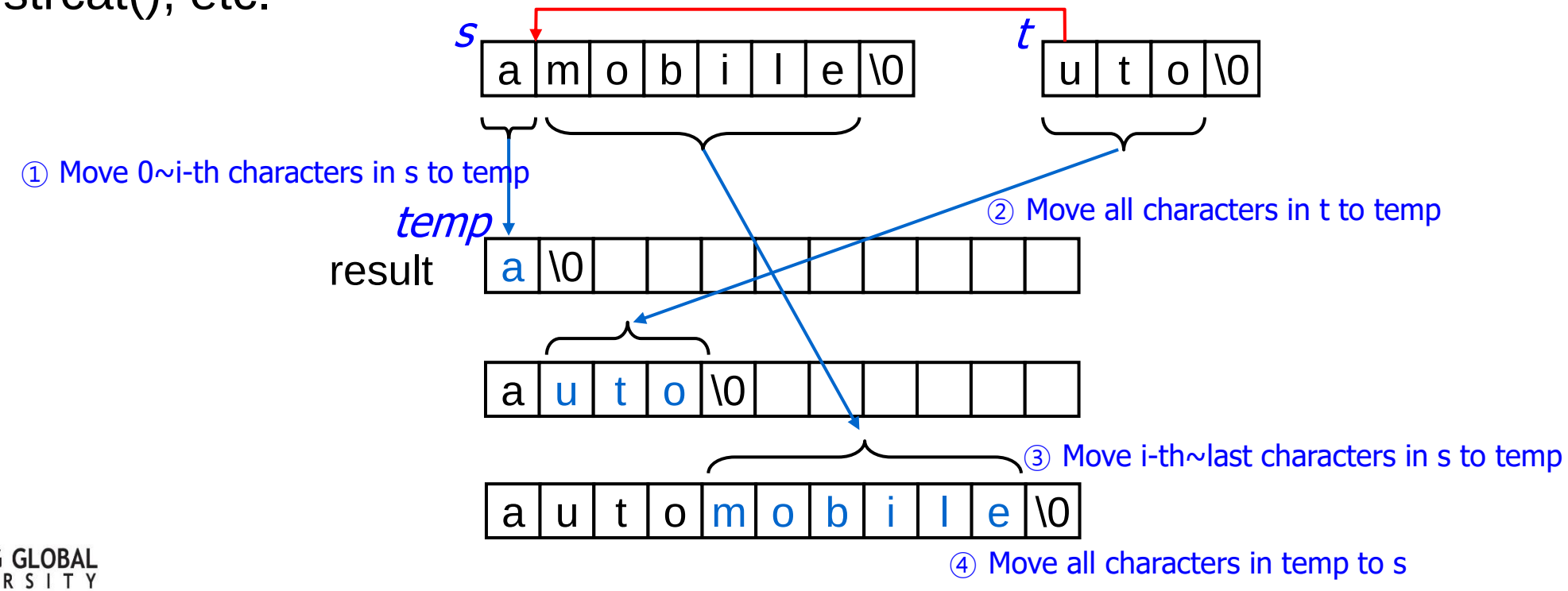
Contacts with Structure

- Expected results

```
PS C:\ds\practice03\sol> .\contacts.exe
Name                    Birthday                E-mail                          Phone Number
-------------------------------------------------------------------------------------------------
Iron Man              / 20210301              / tony@avengers.com             / 111-2222-3333
Spider Man            / 20210401              / peter@avengers.com            / 444-5555-6666
Captain America       / 20210501              / steve@avengers.com            / 777-8888-9999
Black Widow           / 20210601              / natasha@avengers.com          / 123-4567-8901
Thor                  / 20210701              / thor@avengers.com             / 333-5555-7777
Hulk                  / 20210801              / bruce@avengers.com            / 777-8888-9999
```

String Insertion without String Functions

- Implement strnins() without string functions (strcpy, strncpy, strcat).
  - Skeleton code: strnins.c
  - Refer to p.47 ~ p.48, Chapter 2.
  - You can use strlen(), but do not use string functions like strcpy(), strncpy(), strcat(), etc.



String Insertion without String Functions

- Expected results

```
PS C:\ds\practice03\sol> .\strnins.exe
Before strnins(): amobile
After strnins(): amoutobile
PS C:\ds\practice03\sol> .\strnins.exe
Before strnins(): Han University
After strnins(): Handong Global University
```