File-Oriented Input and Output

Instructor: Chien-Ho Ko

Outlines

- **Introduction**
- **Fundamentals of data files**
- **Processing text files**

Introduction

- **Batch input and output**
- **Permanent exist on storage**
- **Without retype data**

Fundamentals of Data Files (1/7)

- **Data file**
  - **A set of records**
- **Record**
  - **A set of data**
  - **Terminate by a new-line character**
- **Data**
  - **Facts**
- **Text and binary files**
  - **Text: store readable and printable characters (*.txt)**
  - **Binary: unreadable characters (*.obj)**
- **End-of-file marker**

Fundamentals of Data Files (2/7)

- **Standard input stream**
  - *stdin* **(standard input devices→program)**
- **Standard output stream**
  - *stdout* **(program→standard output devices)**
- **Standard error stream**
  - *stderr* **(program→standard output devices)**
- **Example (#include<stdio.h>)**
  - *scanf* **(“%d”, &score);**
  - *fscanf* **(stdin, “%d”, &score);**
  - *fscanf* **(file_name, “%d”, &score);**

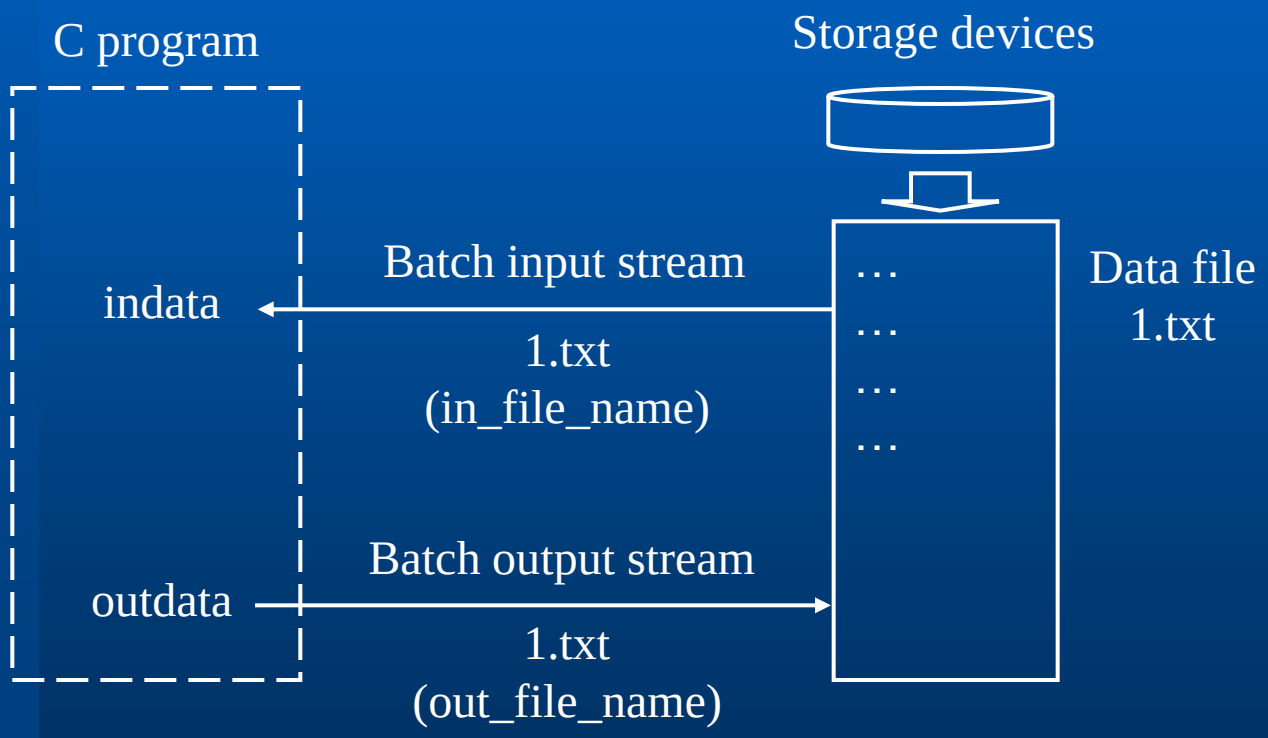
Fundamentals of Data Files (3/7)

- **File declaration**
  - **FILE *file_name;**
- **Internal and external file names**
  - **Internal: file_name**
  - **External: D:\file_name.txt**
- **Open file function:** *fopen*
  - IntFileName=fopen(ExtFileName, OpenMode)

Fundamentals of Data Files (4/7)

- Open mode
  - “r” read data
  - “w” write data
  - “a” append data to the end of file
  - “r+” update: both input and output
  - “w+” update: open a new file for
  - “a+” update: end of file
- Example 15_c01.c

Fundamentals of Data Files (5/7)



Fundamentals of Data Files (6/7)

- **File open verification**
  - **To avoid run-time errors, infinite loop**
  - *exit* (), *#include<stdlib.h>* (standard head file)
  - **Example 15_c02.c**
- **Close file function:** *fclose*
  - **Close files after read/write**
  - **Place an end-of-file marker**
  - **Cut the connection**

Fundamentals of Data Files (7/7)

- **Close file function:** *fclose*
  - **Close before reading it**
  - **Save resources**
  - *fclose(internalFileName);*
  - **Example 15_c03.c**
- **Check for End-of-File:** *feof*
  - *feof(InternalFileName);*
  - **Example 15_c04.c**

Processing Text Files (1/2)

- **Write data to text files**
  - *fprinft(IntFileName, FormatControlString, WriteList);*
  - **Example 15_c05.c**
- **Read data from text files**
  - *fscan(IntFileName, FormatControlString, InputList);*
  - **Example 15_c06.c**

Processing Text Files (2/2)

- **Skipping specific characters in input**
  - **Read the character to be skipped into a variable**
    - **fscanf(file_name, "%4s%c%d", id, &sep, &a);**
  - **Include the character**
    - **fscanf(file_name, "%4s, %d" id, &a);**
  - **Assign suppression character**
    - **fscanf(file_name, "%4s%*c%d" id, &a)**