C Turnin作業3

國立屏東大學 資訊工程系 程式設計(二)

Turnin作業3

- Turnin Code: c.hw3
- Due Date: 2026/03/25 週三 晚上 23:59分截止 **Hard Deadline**

繳交方式說明

本次 Turnin 作業包含多個程式題，建議先為本次 turnin 要繳交的內容建立一個 **外層資料夾**（例如□c.hw3□□切換到該資料夾後再為每一題建立一個 **內層資料夾**（每一題的資料夾名稱已寫於題目前方，例如第一題的資料夾名稱為□p1□□第二題的為□p2□□以此類推），進入到內層資料夾才依照題目要求進行編撰。

同學們可參考如下命令列操作：

```
<ssh 登入系計中後>
[user@ws ~]$ mkdir c.hw3        # 在家目錄建立了一個名為□c.hw3□的資料夾
[user@ws ~]$ cd c.hw3           # 進入□c.hw3□資料夾
[user@ws c.hw3]$ mkdir p1       # 建立一個名為□p1□資料夾
[user@ws c.hw3]$ cd p1          # 切換到□p1□資料夾
[user@ws p1]$ joe reverse.c     # 使用 JOE 編輯器對檔名為 reverse.c 的檔案進行編輯
```

等到我們完成 p1 的撰寫後，請自行加以編譯與執行程式，確認 **正確無誤** 後回到 **外層資料夾** 使用 turnin▲c.hw3▲p1↵ 指令完成 **繳交第一題的整個資料夾**□

```
[user@ws p1]$ cd ..               # 回到上一層資料夾
[user@ws c.hw3]$ turnin c.hw3 p1 # 使用 turnin 指令提交 p1 的程式碼
```

當然，你也可以等到本次作業要求的所有題目都在 c.hw3 資料夾裡完成後，一次將所有在 **目前資料夾中的所有檔案** 都加以上傳。

假設你已經在 c.hw3 資料夾裡完成所有題目，同時確認檔案的繳交格式正確，並且每個題目的程式檔案皆成功編譯並確認執行結果正確後，我們可以使用以下指令將多餘的（不需要繳交的）檔案加以刪除後，一次將所有檔案繳交：

```
[user@ws c.hw3]$ ls                 # 檢視當前資料夾下有哪些內容
p1 p2 p3 p4
[user@ws c.hw3]$ rm -f */a.out      # 移除所有子資料夾中的 a.out 檔案
[user@ws c.hw3]$ turnin c.hw3 .     # 使用 turnin 指令繳交該資料夾下的所有內容
```

```
Turning in:
 ./p6/login.h -- ok
 ./p6/login.c -- ok
 ./p3/search.c -- ok
 ./p3/search.h -- ok
 ./p1/reverse.h -- ok
 ./p1/reverse.c -- ok
 ./p2/isPalindrome.c -- ok
 ./p2/isPalindrome.h -- ok
 ./p7/cardValid.c -- ok
 ./p7/cardValid.h -- ok
 ./p5/trim.h -- ok
 ./p5/trim.c -- ok
 ./p4/numFormat.h -- ok
 ./p4/numFormat.c -- ok
All done.[user@ws c.hw3]$
```

如果繳交後想要查看已繳交的檔案及相關資訊，可以輸入 `turnin▲-ls▲c.hw3↵` 指令，例如：

```
[user@ws c.hw3]$ turnin -ls c.hw3
.:
total 28
drwxrwx---. 2 turninman turnin 4096 Mar 17 19:20 p1
drwxrwx---. 2 turninman turnin 4096 Mar 17 19:20 p2
drwxrwx---. 2 turninman turnin 4096 Mar 17 19:20 p3
drwxrwx---. 2 turninman turnin 4096 Mar 17 19:20 p4
drwxrwx---. 2 turninman turnin 4096 Mar 17 19:20 p5
drwxrwx---. 2 turninman turnin 4096 Mar 17 19:20 p6
drwxrwx---. 2 turninman turnin 4096 Mar 17 19:20 p7

./p1:
total 8
-rw-rw----. 1 turninman turnin 221 Mar 17 19:20 reverse.c
-rw-rw----. 1 turninman turnin  24 Mar 17 19:20 reverse.h

./p2:
total 8
-rw-rw----. 1 turninman turnin 231 Mar 17 19:20 isPalindrome.c
-rw-rw----. 1 turninman turnin  34 Mar 17 19:20 isPalindrome.h

./p3:
total 8
-rw-rw----. 1 turninman turnin 456 Mar 17 19:20 search.c
-rw-rw----. 1 turninman turnin  48 Mar 17 19:20 search.h

./p4:
total 8
-rw-rw----. 1 turninman turnin 664 Mar 17 19:20 numFormat.c
-rw-rw----. 1 turninman turnin  49 Mar 17 19:20 numFormat.h
```

```
./p5:
total 8
-rw-rw----. 1 turninman turnin 565 Mar 17 19:20 trim.c
-rw-rw----. 1 turninman turnin  21 Mar 17 19:20 trim.h

./p6:
total 8
-rw-rw----. 1 turninman turnin 649 Mar 17 19:20 login.c
-rw-rw----. 1 turninman turnin  43 Mar 17 19:20 login.h

./p7:
total 8
-rw-rw----. 1 turninman turnin 1453 Mar 17 19:20 cardValid.c
-rw-rw----. 1 turninman turnin   46 Mar 17 19:20 cardValid.h
[user@ws c.hw3]$
```



本文使用「 」及「 」代表「空白字元」與Enter 換行字元」，並且將使用者輸入的部份使用灰階方式顯示。另外，題目的執行結果中，如果出現「(」、「)」、「:」、「;」、「.」與「,」等符號，皆為英文半形！

本學期作業繳交需要為每一題建立一個資料夾（資料夾名稱為該題題目前方之代號，第一題為p1第二題為p2餘以此類推），繳交方式可參考上述內容，任何未依照正確繳交格式的檔案將以 0 分計。

---

p1 反轉字串

本題需請同學完成一個能夠將使用者輸入的字串進行反轉的函式“ **reverse()**”

請參考下面的 main.c 程式內容：

```
#include <stdio.h>

#include "reverse.h"

int main()
{
    char str[65];
```

```
    printf("Enter a string: ");
    scanf(" %64[^\n]", str);
    str[64] = '\0';

    reverse(str);
    printf("Reversed string: %s\n", str);

    return 0;
}
```

在完成該函式實作後同學需繳交以下檔案：

- **reverse.c**: 此文件包含函式的 **實作**
- **reverse.h**: 此文件包含函式的 **宣告**

另外本題將使用以下 Makefile 進行編譯：

```
all: main.c reverse.o reverse.h
    gcc main.c reverse.o
reverseString.o: reverse.c reverse.h
    gcc -c reverse.c
clean:
    rm -rf *.o *~ *.out
```

執行結果可參考以下的內容：

```
[3:23▲user@ws▲p1]▲./a.out↵
Enter▲a▲string:▲12↵
Reversed▲string:▲21↵
[3:23▲user@ws▲p1]▲./a.out↵
Enter▲a▲string:▲Hello,▲World!↵
Reversed▲string:▲!dlroW▲,olleH↵
[3:23▲user@ws▲p1]▲./a.out↵
Enter▲a▲string:▲!12Hi34Iam%$a78test90#↵
Reversed▲string:▲#09tset87a$%maI43iH21!↵
[3:23▲user@ws▲p1]
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題應繳交檔案如下（**main.c** 與 **Makefile** 則不需繳交）：
  - **reverse.c**
  - **reverse.h**

p2 判斷迴文

迴文指的是一種不管「正著讀」或「反著讀」都完全相同的特殊文句，例如像是 “level” 一詞。

本題需請同學完成一個能夠判斷使用者輸入的字串是否為 **迴文(Palindrome)** 的函式 “ **isPalindrome()**”

請參考以下 main.c 的內容：

```
#include <stdio.h>

#include "isPalindrome.h"

int main()
{
    char str[33];
    printf("Enter a string: ");
    scanf("%[^\n]", str);

    if (isPalindrome(str))
    {
        printf("Is a palindrome.\n");
    }
    else
    {
        printf("Is not a palindrome.\n");
    }

    return 0;
}
```

在完成該函式實作後同學需繳交以下檔案：

- **isPalindrome.c**: 此文件包含函式的 **實作**
- **isPalindrome.h**: 此文件包含函式的 **宣告**

另外本題將使用以下 Makefile 進行編譯：

```
all: main.c isPalindrome.o
    gcc main.c isPalindrome.o
isPalindrome.o: isPalindrome.c
    gcc -c isPalindrome.c
clean:
    rm -rf *.o *~ *.out
```

執行結果可參考以下的內容：

```
[3:23▲user@ws▲p2]▲./a.out↵
Enter▲a▲string:▲radar↵
Is▲a▲palindrome.↵
[3:23▲user@ws▲p2]▲./a.out↵
Enter▲a▲string:▲3r$e$r3↵
Is▲a▲palindrome.↵
[3:23▲user@ws▲p2]▲./a.out↵
Enter▲a▲string:▲Aaa▲a▲aaa↵
Is▲not▲a▲palindrome.↵
[3:23▲user@ws▲p2]
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題應繳交檔案如下（**main.c** 與 **Makefile** 則不需繳交）：
  - **isPalindrome.c**
  - **isPalindrome.h**

---

p3 找子字串

本題需請同學完成一個能夠計算出在一個 **內容字串** 中某一 **目標子字串** 出現的 **次數** 的函式 “ **search()** “。具體來說，從內容字串開頭處進行逐字元的檢查，一旦發現目標子字串時，將次數加 1 並從符合目標子字串內容的下一個字元接續檢查; 反覆進行上述程序，直至內容字串結尾處停止。

舉例來說，如內容字串為 “ **aaaaa** “，且目標子字串為 ” **aa** “，則其出現次數應為 “ **2** ” 次。

請參考以下 main.c 程式碼內容：

```
#include <stdio.h>

#include "search.h"

int main()
{
    char str[1024];
    char substr[16];

    printf("Enter a context: ");
    scanf(" %1023[^\n]", str);
    printf("Enter a substring: ");
```

```
    scanf(" %15[^\n]", substr);

    int count = search(str, substr);
    printf("The substring [%s] ", substr);
    if (count == 0)
    {
        printf("does not appear ");
    }
    else if (count > 1)
    {
        printf("appears %d times ", count);
    }
    else
    {
        printf("appears %d time ", count);
    }
    printf("in the given context.\n");

    return 0;
}
```

在完成該函式實作後同學需繳交以下檔案：

- **search.c**: 此文件包含函式的 **實作**
- **search.h**: 此文件包含函式的 **宣告**

另外本題將使用以下 Makefile 進行編譯：

```
all: main.c search.o
    gcc main.c search.o
trim.o: search.c
    gcc -c search.c
clean:
    rm -rf *.o *~ *.out
```

執行結果可參考以下的內容：

```
[3:23▲user@ws▲p3]▲./a.out↵
Enter▲a▲context:▲aaaaa↵
Enter▲a▲substring:▲aa↵
The▲substring▲[aa]▲appears▲2▲times▲in▲the▲given▲context.↵
[3:23▲user@ws▲p3]▲./a.out↵
Enter▲a▲context:▲cc▲bb▲cc▲dd▲ee▲ff▲gg▲hh▲aa▲bb▲cc▲dd▲ee▲ff▲gg▲hh↵
Enter▲a▲substring:▲aa↵
The▲substring▲[aa]▲appears▲1▲time▲in▲the▲given▲context.↵
```

```
[3:23▲user@ws▲p3]▲./a.out↵
Enter▲a▲context:▲He▲left▲his▲phone▲on▲the▲left▲side▲of▲the▲table,▲then▲walked
▲out▲the▲door▲and▲left▲the▲house▲for▲good.↵
Enter▲a▲substring:▲left↵
The▲substring▲[left]▲appears▲3▲times▲in▲the▲given▲context.↵
[3:23▲user@ws▲p3]▲./a.out↵
Enter▲a▲context:▲The▲quick▲brown▲fox▲jumps▲over▲the▲lazy▲dog.↵
Enter▲a▲substring:▲cat↵
The▲substring▲[cat]▲does▲not▲appear▲in▲the▲given▲context.↵
[3:23▲user@ws▲p3]
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題應繳交檔案如下（**main.c** 與 **Makefile** 則不需繳交）：
  - **search.c**
  - **search.h**

---

p4 手機號碼

台灣的電話號碼的格式通常是 02~09 開頭，其中 02~08 主要保留給市話專用，而剩下的 09 則是保留給行動電話使用，在本題同學需實作一個能夠將使用者所輸入 **連續的10數字** 所組成的手機號碼，格式化成“ **09XX-XXXXXX** ”的函式“ **numFormat()**“

其中，該函式除了將輸入內容格式化外，還需根據以下條件 **由上至下** 判斷：

- 輸入長度是否為“ **10碼**”
- 內容是否只有“ **數字**”
- 開頭前兩碼是否為“ **09**”

若是輸入不符合以上三點條件則必須回以下列列出的回傳值：

- 太長 (回傳 **1**)
- 太短 (回傳 **2**)
- 有數字 **以外** 的內容 (回傳 **3**)
- 開頭兩碼不是 **09** 開始 (回傳 **4**)

請參考以下 main.c 程式碼內容：

```
#include <stdio.h>

#include "numFormat.h"

int main()
{
```

```
    char out[17];
    char in[65];

    printf("Enter a 10-digit number: ");
    scanf(" %64s", in);
    in[64] = '\0';

    int stat = numFormat(in, out, sizeof(out));
    switch (stat)
    {
        case 1:
            printf("The number is too long!\n");
            break;
        case 2:
            printf("The number is too short!\n");
            break;
        case 3:
            printf("The number has invalid characters!\n");
            break;
        case 4:
            printf("The number must start with 09!\n");
            break;
        case 0:
            printf("Result: %s\n", out);
    }

    return stat;
}
```

在完成該函式實作後同學需繳交以下檔案：

- **numFormat.c**: 此文件包含函式的 **實作**
- **numFormat.h**: 此文件包含函式的 **宣告**

另外本題將使用以下 Makefile 進行編譯：

```
all: main.c numFormat.o numFormat.h
    gcc main.c numFormat.o
numFormat.o: numFormat.c numFormat.h
    gcc -c numFormat.c
clean:
    rm -rf *.o *~ *.out
```

本題的執行結果可參考以下的內容：

```
[3:23▲user@ws▲p4]▲./a.out↵
Enter▲a▲10-digit▲number:▲0912345678↵
Result:▲0912-345678↵
[3:23▲user@ws▲p4]▲./a.out↵
Enter▲a▲10-digit▲number:▲0193849810273↵
The▲number▲is▲too▲long!↵
[3:23▲user@ws▲p4]▲./a.out↵
Enter▲a▲10-digit▲number:▲19843↵
The▲number▲is▲too▲short!↵
[3:23▲user@ws▲p4]▲./a.out↵
Enter▲a▲10-digit▲number:▲hij094JO)0↵
The▲number▲has▲invalid▲characters!↵
[3:23▲user@ws▲p4]▲./a.out↵
Enter▲a▲10-digit▲number:▲1234567890↵
The▲number▲must▲start▲with▲09!↵
[3:23▲user@ws▲p4]
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題應繳交檔案如下（**main.c** 與 **Makefile** 則不需繳交）：
  - **numFormat.c**
  - **numFormat.h**

---

p5 去頭去尾

本題需請同學完成一個能夠去除使用者在 **頭尾** 額外輸入的 **空白(Whitespace)** 的函式 “ **trim()**”

```
#include <stdio.h>

#include "trim.h"

int main()
{
    char str[33];

    printf("Enter a string:");
    fgets(str, sizeof(str), stdin);
    for (int i = 0; str[i] != '\0'; i++)
    {
        if (str[i] == '\n')
        {
            str[i] = '\0';
            break;
```

```
        }
    }

    printf("Before: [%s]\n", str);
    trim(str);
    printf("After: [%s]\n", str);

    return 0;
}
```

在完成該函式實作後同學需繳交以下檔案：

- **trim.c**: 此文件包含函式的 **實作**
- **trim.h**: 此文件包含函式的 **宣告**

另外本題將使用以下 Makefile 進行編譯：

```
all: main.c trim.o
    gcc main.c trim.o
trim.o: trim.c
    gcc -c trim.c
clean:
    rm -rf *.o *~ *.out
```

本題的執行結果可參考以下的內容：

```
[3:23▲user@ws▲p1]▲./a.out↵
Enter▲a▲string:▲▲▲▲Hello,World!▲▲▲▲▲↵
Before:▲[▲▲▲▲Hello,World!▲▲▲▲▲]↵
After:▲[Hello,World!]↵
[3:23▲user@ws▲p1]▲./a.out↵
Enter▲a▲string:▲▲▲▲▲▲▲I▲am▲a▲student.▲▲▲▲▲▲▲▲▲↵
Before:▲[▲▲▲▲▲▲▲I▲am▲a▲student.▲▲▲▲▲▲▲▲]↵
After:▲[I▲am▲a▲student.]↵
[3:23▲user@ws▲p1]▲./a.out↵
Enter▲a▲string:▲▲▲H*(.▲G^H▲▲UO▲I1▲2▲▲▲jlk▲23;▲▲▲↵
Before:▲[▲▲▲H*(.▲G^H▲▲UO▲I1▲2▲▲▲jlk▲23;▲▲]↵
After:▲[H*(.▲G^H▲▲UO▲I1▲2▲▲▲jlk▲23;]↵
[3:23▲user@ws▲p1]
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題應繳交檔案如下（**main.c** 與 **Makefile** 則不需繳交）：



- **trim.c**
- **trim.h**

---

p6 帳密登入

本題需請同學實作一個簡單的登入函式“ **login()**“

其中該函式需要有以下資料來去驗證使用者的登入資訊：

帳號	密碼
junwu	Professor Wu
someone	at Some Where
linda	Smart Girl
markonbizz	Mark's Password

另外，該函式除了判斷以上表格資訊外，還需判斷以下條件：

- 帳號是否存在 (若不存在，回傳“ **1**”)
- 密碼是否錯誤 (若錯誤，回傳“ **2**”)

請參考以下 main.c 程式碼內容：

```
#include <stdio.h>

#include "login.h"

int main()
{
    char acc[20];
    char pw[20];

    printf("Account? ");
    scanf(" %[^\n]", acc);
    printf("Password? ");
    scanf(" %[^\n]", pw);

    int stat = login(acc, pw);

    switch (stat)
    {
        case 1:
            printf("Account does not exist!\n");
            break;
        case 2:
            printf("Wrong password!\n");
```

```
            break;
        default:
            printf("Login successful\n");
    }

    return stat;
}
```

在完成該函式實作後同學需繳交以下檔案：

- **login.c**: 此文件包含函式的 **實作**
- **login.h**: 此文件包含函式的 **宣告**

另外本題將使用以下 Makefile 進行編譯：

```
all: main.c login.o
    gcc main.c login.o
login.o: login.c login.h
    gcc -c login.c
clean:
    rm -rf *.o *~ *.out
```

本題的執行結果可參考以下的內容：

```
[3:23▲user@ws▲p6]▲./a.out↵
Account?▲markonbizz↵
Password?▲Mark's▲Password↵
Login▲successful↵
[3:23▲user@ws▲p6]▲./a.out↵
Account?▲linus↵
Password?▲123456789↵
Account▲does▲not▲exist!↵
[3:23▲user@ws▲p6]▲./a.out↵
Account?▲someone↵
Password?▲at▲Some▲place↵
Wrong▲password!↵
[3:23▲user@ws▲p6]
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題應繳交檔案如下（**main.c** 與 **Makefile** 則不需繳交）：
  - **login.c**
  - **login.h**

p7 檢查信用卡

信用卡、簽帳金融卡等「塑膠貨幣」，主要用於線上購物或是身上現金不足時使用，本題需請同學設計一個專門驗證“ **16碼** ”的信用卡驗證函式“ **cardValid()** ”來驗證使用者輸入的卡號。

該函式主要分成兩個步驟：

1. 卡號是否正確
2. 該卡是由哪個 **發卡機構** 發行

在“ **卡號是否正確** ”首先會判斷以下條件：

1. 長度是否為“ **16碼** “，若長度不為16，則回傳“ **0** “，否之則繼續下一步。
2. 是否含有“ **非數字** ”字元，若含有非數字字元，則回傳“ **0** “，否之則繼續下一步。
3. 透過 **Luhn算法** 檢查，方法請見下面區塊



Luhn算法

1. 先 **由右到左** 的順序將 **偶數位數字** 乘以2，若有數字計算後 $\gt 9$則該數字需再減去 9
2. 以上一步 **同樣的方向**，將連同未經處理過的“ **奇數位數字** ”跟 處理過的“ **偶數位數字** ”總加起來
3. 最後將該“ **總和 模10$(mod 10)$ 計算** “，並判斷結果是否為“ **0** ”, 若等於“ **0** “，則該卡號驗證成功，否則失敗

若該卡號經判定有效後，請接著判斷其發行者代碼：

- **Visa**：第一位以 4 開頭，若符合請回傳“ **1** ”
- **Mastercard**：前 2 位以 51～55 開頭，若符合請回傳“ **2** ”
- **JCB**：前 4 位以 3528～3589 開頭，若符合請回傳“ **3** ”
- **上述以外機構**：回傳“ **0** ”

請參考以下 main.c 程式碼內容：

```
#include <stdio.h>

#include "cardValid.h"

int main()
{
    char number[33];
    int valid = 0;
    int res = 0;

    printf("Enter credit card number: ");
    scanf(" %16s", number);
```

```
    res = cardValid(number, &valid);

    if (valid == 0)
    {
        printf("Invalid card number!\n");
        return 1;
    }

    switch (res)
    {
        case 1:
            printf("It's Visa!\n");
            break;
        case 2:
            printf("It's MasterCard\n");
            break;
        case 3:
            printf("It's JCB\n");
            break;
        default:
            printf("Unknown bank id!\n");
    }

    return 0;
}
```

在完成該函式實作後同學需繳交以下檔案：

- **cardValid.c**: 此文件包含函式的 **實作**
- **cardValid.h**: 此文件包含函式的 **宣告**

另外本題將使用以下 Makefile 進行編譯：

```
all: main.c cardValid.o
    gcc main.c cardValid.o
cardValid.o: cardValid.c cardValid.h
    gcc -c cardValid.c
clean:
    rm -rf *.o *~ *.out
```

本題的執行結果可參考以下的內容：

```
[3:23▲user@ws▲p7]▲./a.out↵
Enter▲credit▲card▲number:▲4032039829398173↵
It's▲Visa!↵
```

```
[3:23▲user@ws▲p7]▲./a.out↵
Enter▲credit▲card▲number:▲5038225257429561↵
Unknown▲bank▲id!↵
[3:23▲user@ws▲p7]▲./a.out↵
Enter▲credit▲card▲number:▲4123456789012344↵
Invalid▲card▲number!↵
[3:23▲user@ws▲p7]▲./a.out↵
Enter▲credit▲card▲number:▲12345↵
Invalid▲card▲number!↵
[3:23▲user@ws▲p7]▲./a.out↵
Enter▲credit▲card▲number:▲123451234512345123451234512345↵
Invalid▲card▲number!↵
[3:23▲user@ws▲p7]▲./a.out↵
Enter▲credit▲card▲number:▲4b234c678$01234a↵
Invalid▲card▲number!↵
[3:23▲user@ws▲p7]
```



- 本題相關的程式碼路徑已註明於檔名右側，同學們可以透過路徑複製到自己的家目錄。
- 本題應繳交檔案如下（**main.c** 與 **Makefile** 則不需繳交）：
  - **cardValid.c**
  - **cardValid.h**

From:
https://junwu.nptu.edu.tw/dokuwiki/ - **Jun Wu的教學網頁**
國立屏東大學資訊工程學系
CSIE, NPTU
Total: 334357

Permanent link:
https://junwu.nptu.edu.tw/dokuwiki/doku.php?id=c:2026spring:hw3

Last update: **2026/03/22 12:38**