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: 5/13 00:00 **Hard Deadline**

繳交方式說明

本次Turnin作業包含多個程式題，請為每一題建立一個資料夾，並將該題所要上傳的檔案放置其中後，再使用turnin指令上傳作業。請同學先為本次作業建立一個資料夾hw3，然後在hw3裡分別為每一題建立一個子資料夾，用以進行每一題的作答以及上傳。每一題的子資料夾名稱已寫於題目前方，請務必依照題目的規定建立子資料夾，例如第1題為p1，第2題為p2，餘依此類推。當我們完成某一個題目的作答後，就可以使用turnin指令將該題的答案上傳。以第1題為例，當我們在p1子資料夾裡完成作答後，就可以回到hw3資料夾，使用以下指令將其上傳：

[3:23 user@ws hw3] `turnin▴c.hw3▴p1↵`

當然，你也可以等到所有題目都完成後，回到hw3資料夾，使用以下指令將所有題目都加以上傳：

[3:23 user@ws hw2] `turnin▴c.hw3▴.↵`

註：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。

---

p1 聯絡人

請設計一個程式用以管理 10 個聯絡人資訊，其中每個聯絡人包含以下資訊:

- name(姓名)
- gender(性別)
- birthday(生日)
- phonetype(電話型態)
- phone(電話號碼)
- address(住址)

請參考以下的 Contact.h所定義的相關結構體、列舉資料型別與共有體:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

```
#define numContact 10

typedef enum {Male, Female} Gender;

typedef enum {January, February, March, April, May, June, July,
              August, September, October, November, December} Month;

typedef struct
{
  Month month;
  short day;
  short year;
} Date;

typedef struct
{
  char firstname[20];
  char lastname[10];
} Name;
typedef enum {CHT, TWN, FET} Carrier;

typedef struct
{
  char number[10];
  Carrier carrier;
} Mobile;

typedef struct
{
  char areacode[4];
  char number[8];
} Landline;

typedef enum {LandLine, MobilePhone} PhoneType;

typedef struct
{
  Name name;
  Gender gender;
  Date birthday;
  PhoneType phonetype;
  union
  {
    Landline landline;
    Mobile mobile;
  } phone;
  char address[50];
} Contact;

Contact getAContact();
void showAContact(Contact c);
```

```
void sortContacts(Contact cs[]);
```

請設計並實作一個名為 Contact.c 的程式，並在其中完成getAContact()、showAContact()與「sortContacts()函式之實作，其中sortContacts()函式是以聯絡人的年齡進行排序，年紀愈輕的排在愈前面(換句話說，出生日期愈早的人年紀愈大，排在愈後面)。相關程式功能可以使用以下的Main.c進行測試:

```
#include "Contact.h"

int main()
{
  int i;

  Contact mycontacts[numContact];
  for(i=0;i<numContact;i++)
    mycontacts[i]=getAContact();

  sortContacts(mycontacts);

  for(i=0;i<numContact;i++)
    showAContact(mycontacts[i]);
}
```

本題的相關程式將使用以下的Makefile進行編譯：

```
all: Main.c Contact.o
    cc Main.c Contact.o

Contact.o: Contact.c Contact.h
    cc -c Contact.c

clean:
    rm -f *.o *~ *.*~ a.out
```

此題的執行結果如下：

```
[3:23 user@ws hw] ./a.out↵
Name:▲Amy▲Wang↵
Gender▲(M/F):▲F↵
Birthday▲(YYYY/MM/DD):▲1977/3/5↵
Phone▲Type▲(L/M):▲L↵
Number:▲(08)7238700↵
Address:▲No.51,▲Mingsheng▲E.Rd.,▲Pingtung▲City↵
```

```
↵
Name:▲Kung▲Liu↵
Gender▲(M/F):▲M↵
Birthday▲(YYYY/MM/DD):▲1995/12/3↵
Phone▲Type▲(L/M):▲M↵
Number:▲0918123456↵
Carrier▲(C/T/F):▲T↵
Address:▲No.99,▲ChongHui▲St.,▲Taipei▲City↵
↵
□□⋮
Kung▲Liu▲(Male)▲December▲3rd,▲1995,▲0918123456(Taiwan▲Mobile),▲No.99,▲ChongHui▲St.,▲Taipei▲City↵
Amy▲Wang▲(Female)▲March▲5th,▲1977,▲(08)7238700,▲No.51,▲Mingsheng▲E.Rd.,▲Pingtung▲City↵
□□⋮
[3:23 user@ws hw]
```

- 注意1：本題只需繳交Contact.c□其它檔案則不需繳交。
- 注意2：在電信業者方面，中華電信以Chunghwa▲Telecom表示，遠傳以 FarEasTone表示，台灣大哥大以Taiwan▲Mobile表示。

p2 紀念品商品管理

課本「程式演練21」的程式式經部份修改後，已放置於/home/stu/public/c2024s/hw3/p2資料夾內，請同學取得並再針對以下需求進行修改：

1. 由於測試版僅提供10筆商品資料的操作，但目前除了使用□Product products[10];□宣告大小為10的Product結構體陣列外，並沒有針對當使用者新增超過10筆商品資料時的異常處理。因此目前的測試版當使用者輸入過多資料時，此程式的執行會發生異常，且無任何的錯誤訊息供使用者瞭解狀況。請試著修改相關程式來解決此一問題，並在發生超過10筆商品資料時，顯示□Can't▲insert▲more▲than▲10▲products!□錯誤訊息。
2. 請試著修改相關程式來增加一個□d□指令，詢問使用者所欲刪除的商品之代碼□ID□□並在已經輸入的商品資訊中尋找符合的商品並將其加以移除。
3. 請試著修改相關程式來增加□b□□□k□與「t□指令，分別用以將所有書籍、鑰匙圈與紀念T恤的商品清單輸出。
4. 請修改此程式的輔助說明(指令h)□使其能顯示以上所新增的指令。

本題的執行結果可參考如下：

```
[9:19 user@ws hw] ./a.out↵
command?h↵
b:▲list▲all▲book▲products.↵
d:▲delete▲a▲product.↵
i:▲insert▲a▲new▲product.↵
k:▲list▲all▲key▲chain▲products.↵
l:▲list▲all▲products.↵
s:▲search▲for▲a▲product.↵
t:▲list▲all▲T-shirt▲products.↵
q:▲quit.↵
```

```
command?i↵
ID=?101↵
price=?100↵
type (b for book, k for keychain, t for T-shirt)=?b↵
Author?Frank▲Gallagher↵
command?i↵
ID=?102↵
price=?150.5↵
type (b for book, k for keychain, t for T-shirt)=?b↵
Author?Kevin▲Bald↵
⋮
command?i↵
Can't▲insert▲more▲than▲10▲products!↵
command?l↵
ID:101▲price=100.00▲Book(▲Frank▲Gallagher▲)↵
ID:102▲price=150.50▲Book(▲Kevin▲Bald▲)↵
ID:103▲price=168.99▲Book(▲Scott▲Campbell▲)↵
ID:104▲price=66.80▲Keychain (▲steel▲)↵
ID:105▲price=34.20▲Keychain(▲copper▲)↵
ID:106▲price=55.30▲Keychain(▲wood▲)↵
ID:107▲price=23.50▲Keychain(▲wood▲)↵
ID:108▲price=88.20▲T-Shirt(▲M▲)↵
ID:109▲price=200.00▲T-Shirt(▲XL▲)↵
ID:110▲price=99.50▲T-Shirt(▲M▲)↵
command?b↵
ID:101▲price=100.00▲Book(▲Frank▲Gallagher▲)↵
ID:102▲price=150.50▲Book(▲Kevin▲Bald▲)↵
ID:103▲price=168.99▲Book(▲Scott▲Campbell▲)↵
command?k↵
ID:104▲price=66.80▲Keychain (▲steel▲)↵
ID:105▲price=34.20▲Keychain(▲copper▲)↵
ID:106▲price=55.30▲Keychain(▲wood▲)↵
ID:107▲price=23.50▲Keychain(▲wood▲)↵
command?d↵
ID?106↵
command?k↵
ID:104▲price=66.80▲Keychain (▲steel▲)↵
ID:105▲price=34.20▲Keychain(▲copper▲)↵
ID:107▲price=23.50▲Keychain(▲wood▲)↵
command?d↵
ID?106↵
The▲product▲not▲found!↵
command?t↵
ID:108▲price=88.20▲T-Shirt(▲M▲)↵
ID:109▲price=200.00▲T-Shirt(▲XL▲)↵
ID:110▲price=99.50▲T-Shirt(▲M▲)↵
command?q↵
[9:19 user@ws hw]
```

- 注意1：本題可繳交ProductInfo.c、ProductInfo.h與SouvenirShopMain.c、至於Makefile則不需繳交。
- 注意2：為了讓同學更清楚此題的輸出的結果，於/home/stu/public/c2024s/hw3/p2裡，另外提供一個test檔案。此test檔案為本題的可執行檔，同學可加以執行。

p3 五子棋程式設計(活四)

請參考課本「程式演練22」，請修改程式、增加功能，讓程式可以找出棋盤中所有活四的位置。所謂的「活四」是指已經完成四子連線(縱、橫或斜向的連續四個相同顏色的棋子)，且其前後兩端都沒有棋子，只要任意在其兩端中擇一加入同顏色的棋子即可完成五子連線。為簡化起見，本題不需找出洞四，也就是在連續五個位置中，除頭尾外中間某個位置留空，但另外四子為同一方的棋子的情況(當然洞四的頭尾外側也沒有對手防堵的棋子)。此題的輸出格式可參考如下:

```
[9:19 user@ws hw] ./a.out▲<▲chess.1↵
Black▲live▲fours▲are:↵
none↵
White▲live▲fours▲are:↵
none↵
[9:19 user@ws hw] ./a.out▲<▲chess.2↵
Black▲live▲fours▲are:↵
<F,▲6>,▲<F,▲7>,▲<F,▲8>▲and▲<F,▲9>↵
White▲live▲fours▲are:↵
<M,▲11>,▲<N,▲11>,▲<O,▲11>▲and▲<P,▲11>↵
<H,▲8>,▲<I,▲9>,▲<J,▲10>▲and▲<K,▲11>↵
[9:19 user@ws hw]
```

本題不限定所繳交的程式檔名，但必須繳交 Makefile、並將可執行檔編譯為a.out;另外，你可以在/home/stu/public/c2024s/hw3/p3資料夾裡找到chess.1與chess.2測試檔。 注意：所找到的活四只需依黑子、白子的順序加以輸出，但組成活四的棋子位置輸出可依任意順序，且超過一個以上 活四亦不需要排序其先後順序。但棋子位置間的逗號、空白、大於、小於等符號必須依照前述的執行結果。

p4 五子棋程式設計(死四)

承上題，請參考課本「程式演練22」，請修改程式、增加功能，讓程式可以找出棋盤中所有死四的位置。所謂的「死四」是指已經完成四子連線(縱、橫或斜向的連續四個相同顏色的棋子)，且其前後兩端有一端有對手的棋子。同樣為簡化起見，本題不需找出洞死四，也就是在連續五個位置中，除頭尾外中間某個位置留空，但另外三子為同一方的棋子的情況(當然洞死四的頭尾某側已有對手防堵的棋子)。此題的輸出格式可參考如下:

```
[9:19 user@ws hw] ./a.out▲<▲chess.1↵
Black▲dead▲fours▲are:↵
none↵
White▲dead▲fours▲are:↵
none↵
[9:19 user@ws hw] ./a.out▲<▲chess.2↵
Black▲dead▲fours▲are:↵
```

```
<F,▲7>,▲<F,▲8>,▲<F,▲9>▲and▲<F,▲10>↵
White▲dead▲fours▲are:↵
<I,▲9>,▲<J,▲10>,▲<K,▲11>▲and▲<L,▲12>↵
<A,▲1>,▲<A,▲2>,▲<A,▲3>▲and▲<A,▲4>↵
[9:19 user@ws hw]
```

本題不限定所繳交的程式檔名，但必須繳交 Makefile並將可執行檔編譯為a.out;另外，你可以在/home/stu/public/c2024s/hw3/p4資料夾裡找到chess.1與chess.2測試檔。 注意：所找到的死四只需依黑子、白子的順序加以輸出，但組成死四的棋子位置輸出可依任意順序，且超過一個以上死四亦不需要排序其先後順序。但棋子位置間的逗號、空白、大於、小於等符號必須依照前述的執行結果。

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: 241548

Permanent link:
https://junwu.nptu.edu.tw/dokuwiki/doku.php?id=c:2024spring-hw3

Last update: **2024/05/06 13:37**