國立屏東大學 資訊工程系 程式設計(一)

Turnin作業2

- Turnin Code: c.hw2
- Due Date: 2025/09/30 23:59 **Hard Deadline**

繳交方式說明

本次Turnin作業包含6個程式題，建議可以為它們建立一個資料夾，並將此次作業所有要上傳的檔案放置於該資料夾後，再使用 `turnin` 指令上傳作業。例如下面的操作在同學的家目錄裡建立了一個名為hw2的資料夾，並且切換到該資料夾後，針對第1題所要求的MemAddress.c進行程式編撰：

```
[user@ws ~]$ mkdir hw2
[user@ws ~]$ cd hw2
[user@ws hw2]$ joe MemAddress.c
```

等到我們完成MemAddress.c的撰寫後，請自行加以編譯與執行程式，確認正確後使用 `turnin␣c.hw2␣MemAddress.c↵` 指令將加以上傳：

```
[user@ws c.hw2]$ turnin c.hw2 MemAddress.c
Turning in:
 DoMath.c -- ok
All done.
[user@ws c.hw2]$
```

當然，你也可以等到本次作業要求的所有題目都在hw2資料夾裡完成後，再使用 `turnin␣c.hw2␣.↵` 指令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hw2資料夾裡撰寫完所有題目，並且每個題目的程式檔案皆已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.hw2]$ ls
a.out              MemSizeOfFloatings.c  SquareMeter2Ping.c
ASCIIShift.c       NumberConverter.c
MemAddress.c       Rounded.c
[user@ws c.hw2]$ rm a.out
[user@ws c.hw2]$ ls
ASCIIShift.c       MemSizeOfFloatings.c  Rounded.c
MemAddress.c       NumberConverter.c     SquareMeter2Ping.c
[user@ws c.hw2]$ turnin c.hw2 .
Turning in:
```

```
 ./SquareMeter2Ping.c -- ok
 ./Rounded.c -- ok
 ./NumberConverter.c -- ok
 ./MemSizeOfFloatings.c -- ok
 ./MemAddress.c -- ok
 ./ASCIIShift.c -- ok
All done.
[user@ws c.hw2]$
```

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.hw2↵ 指令，例如：

```
[user@ws c.hw2]$ turnin -ls c.hw2
.:
total 24
-rw-rw----. 1 turninman turnin 162 Sep 14 20:26 ASCIIShift.c
-rw-rw----. 1 turninman turnin 240 Sep 14 20:26 MemAddress.c
-rw-rw----. 1 turninman turnin 207 Sep 14 20:26 MemSizeOfFloatings.c
-rw-rw----. 1 turninman turnin 349 Sep 14 20:26 NumberConverter.c
-rw-rw----. 1 turninman turnin 289 Sep 14 20:26 Rounded.c
-rw-rw----. 1 turninman turnin 259 Sep 14 20:26 SquareMeter2Ping.c
[user@ws c.hw2]$
```

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。另外，出現在程式執行結果中的符號皆為半形。

P1 記憶體位置（MemAddress.c）

請設計一個C語言的程式MemAddress.c，宣告一個int型態的變數x，並將其所分配到的記憶體位址（包含開始的位元組位址到結束的位元組位址）輸出，其執行結果可參考以下的畫面：

```
[7:17 user@ws hw2] ./a.out↵
An␣integer␣is␣located␣at␣0x7ffda2f128f4-0x7ffda2f128f7.↵
[7:17 user@ws hw2]
```

【提示：上述執行結果**僅供參考**，變數實際分配到的記憶體位址將依作業系統分配而定，每次執行結果並不相同。你可以使用&x來取得x所分配到的記憶體空間的起始位址，再利用sizeof(int)或sizeof(x)來取得一個int整數變數所占用的記憶體空間為多少後，計算出所分配到的記憶體空間的結束位址為何。】

【提示：本題若直接輸出「固定」的結果將不予計分。】

P2 浮點數型別之記憶體大小（MemSizeOfFloatings.c）

請設計一個C語言的程式MemSizeOfFloatings.c，將float、double與long double等浮點數型態所佔用的記憶

體空間（單位為位元組）輸出，其執行結果可參考以下的畫面：

```
[3:23 user@ws hw2] ./a.out↵
The▲size▲of▲a▲float▲variable▲is▲4▲bytes.↵
The▲size▲of▲a▲double▲variable▲is▲8▲bytes.↵
The▲size▲of▲a▲long▲double▲variable▲is▲16▲bytes.↵
[3:23 user@ws hw2]
```



所有作業都會在ws上進行批改，所以此題請不要「直接輸出以上的結果」，而是必須要能夠依據不同的作業環境以及編譯器輸出的正確的資料型態大小。若直接輸出以上的結果將不予計分。

P3 數字系統轉換（NumberConverter.c）

請設計一個C語言的程式NumberConverter.c，要求使用者輸入一個10進制與一個8進制的整數，並將這兩個整數轉換為16進制後輸出，其執行結果可參考以下的畫面：

```
[3:23 user@ws hw2] ./a.out↵
Please▲input▲an▲integer▲in▲decimal:▲100↵
(100)_10▲=▲(64)_16↵
↵
Please▲input▲an▲integer▲in▲octal:▲77↵
(77)_8▲=▲(3f)_16↵
[3:23 user@ws hw2]
```

P4 土地單位轉換（SquareMeter2Ping.c）

請設計一個C語言程式SquareMeter2Ping.c，讓使用者以平方公尺為單位輸入一筆土地面積（可包含小數），將其轉換為對應的坪數（台灣慣用的面積單位）後輸出。注意，本題如果有使用浮點數的需求，請一律使用 **double型態**，並以 **1平方公尺＝0.3025坪** 進行計算。本題的執行結果可參考以下的畫面：

```
[3:23 user@ws hw2] ./a.out↵
Please▲input▲an▲area▲in▲square▲meters:▲100↵
100.000000▲square▲meters▲=▲30.250000▲ping.↵
[3:23 user@ws hw2] ./a.out↵
Please▲input▲an▲area▲in▲square▲meters:▲12.5↵
12.500000▲square▲meters▲=▲3.781250▲ping.↵
```

```
[3:23 user@ws hw2]
```

P5 四捨五入（Rounded.c）

請設計一個C語言程式Rounded.c，接受使用者所輸入的一個非負的浮點數(請使用double型態)，將其小數部份四捨五入後輸出，其執行結果可參考如下：

```
[7:17 user@ws hw2] ./a.out↵
Floating▲point▲number:▲13.14↵
The▲rounded▲value▲is▲13.↵
[7:17 user@ws hw2] ./a.out↵
Floating▲point▲number:▲38.5832↵
The▲rounded▲value▲is▲39.↵
[7:17 user@ws hw2] ./a.out↵
Floating▲point▲number:▲8↵
The▲rounded▲value▲is▲8.↵
[7:17 user@ws hw2]
```

P6* 字元位移（ASCIIShift.c）

請設計一個C語言的程式ASCIIShift.c，要求使用者輸入一個字元與偏移量，並將該字元進行指定的偏移後輸出其值。要特別注意的是，本題所輸入的字元只會是小寫英文字母，且偏移量只會輸入大於等於0的整數。具體來說，小寫的英文字母依序包含a、b、c、...、x、y到z共26個字母，當某個字母要進行偏移時，即表示要往其後面移動若干個位置。例如a偏移3位為d，因為a往後移動3個位置，就是從a變成b、b變成c然後c再變成d，共移動3次位置。但是當移動到超出最後一個小寫英文字母z後，則請移回第一個小寫英文字母a繼續移動，例如x偏移4位為b，因為x往後移動4個位置，就是從x變成y、y變成z、z變成a(因為超出z後就必須移回a)，然後a再變成b，共移動4次位置。本題的執行結果可參考以下的畫面：

```
[3:23 user@ws hw2] ./a.out↵
Please▲input▲a▲character▲and▲an▲offset:a▲5↵
'a'(97)▲shift▲5▲is▲'f'(102).↵
[3:23 user@ws hw2] ./a.out↵
Please▲input▲a▲character▲and▲an▲offset:b▲26↵
'b'(98)▲shift▲26▲is▲'b'(98).↵
[3:23 user@ws hw2] ./a.out↵
Please▲input▲a▲character▲and▲an▲offset:b▲28↵
'b'(98)▲shift▲28▲is▲'d'(100).↵
[3:23 user@ws hw2] ./a.out↵
Please▲input▲a▲character▲and▲an▲offset:y▲5↵
'y'(121)▲shift▲5▲is▲'d'(100).↵
[3:23 user@ws hw2]
```

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: 334337

Permanent link:
https://junwu.nptu.edu.tw/dokuwiki/doku.php?id=c:2025fall-hw2

Last update: **2025/09/18 13:36**