2025/11/19 11:22 1/13 Turnin作業7

國立屏東大學 資訊工程系 程式設計(一)

Turnin作業7

• chapter 7

• Turnin Code: c.hw7

• Due Date: 2025/11/25 23:59 Hard Deadline

繳交方式說明

本次Turnin作業包含9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hw7的資料夾,並且切換到該資料夾後,針對第1題所要求的LCM.c進行程式編撰:

```
[user@ws ~]$ mkdir hw7
[user@ws ~]$ cd hw7
[user@ws hw7]$ joe LCM.c
```

等到我們完成LCM.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.hw7▲ LCM.c↓ 指令將加以上傳:

```
[user@ws c.hw7]$ turnin c.hw7 LCM.c
Turning in:
  LCM.c -- ok
All done.
[user@ws c.hw7]$
```

當然,你也可以等到本次作業要求的所有題目都在hw7資料夾裡完成後,再使用 turnin Ac. hw7 A. J. 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hw7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.hw7]$ ls
Factorial.c
                                                        PrimeNumberSum.c
                            LCM.c
                                                        PrintDiamond.c
GCD.c
                            NumberPyramidAlignment.c
HollowInvertedTriangle.c
                            NumPalindromePyramid.c
RangeMultiplicationTable.c
a.out
[user@ws c.hw7]$ rm a.out
[user@ws c.hw7]$ ls
Factorial.c
                            LCM.c
                                                        PrimeNumberSum.c
```

```
GCD.c
                            NumberPyramidAlignment.c
                                                        PrintDiamond.c
HollowInvertedTriangle.c
                            NumPalindromePyramid.c
RangeMultiplicationTable.c
[user@ws c.hw7]$ turnin c.hw7 .
Turning in:
 ./Factorial.c -- ok
 ./LCM.c -- ok
 ./PrimeNumberSum.c -- ok
 ./HollowInvertedTriangle.c -- ok
 ./NumberPyramidAlignment.c -- ok
 ./PrintDiamond.c -- ok
 ./HollowInvertedTriangle.c -- ok
 ./NumPalindromePyramid.c -- ok
 ./RangeMultiplicationTable.c -- ok
All done.
[user@ws c.hw7]$
```

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.hw7→ 指令,例如:

```
[user@ws c.hw7] c.hw7 $ turnin -ls c.hw7
. :
total 40
-rw-rw----. 1 turninman turnin 922 Nov
                                        18 3:25 Factorial.c
-rw-rw----. 1 turninman turnin 506 Nov
                                        18 3:25 GCD.c
-rw-rw----. 1 turninman turnin 468 Nov
                                        18 3:25 HollowInvertedTriangle.c
-rw-rw----. 1 turninman turnin 1418 Nov
                                        18 3:25 LCM.c
                                        18 3:25 NumberPyramidAlignment.c
-rw-rw----. 1 turninman turnin 549 Nov
-rw-rw----. 1 turninman turnin 592 Nov
                                        18 3:25 NumPalindromePyramid.c
-rw-rw----. 1 turninman turnin 624 Nov
                                        18 3:25 PrimeNumberSum.c
-rw-rw----. 1 turninman turnin 2876 Nov
                                        18 3:25 PrintDiamond.c
-rw-rw----. 1 turninman turnin 508 Nov
                                        18 3:25 RangeMultiplicationTable.c
```

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 最小公倍數□LCM.c□

題目敘述:

請設計一個 C 語言程式[]LCM.c[]]讓使用者輸入兩個正整數,計算並輸出這兩個數字的最小公倍數[]LCM, Least Common Multiple[]後加以輸出。假設使用者所輸入的兩個整數分別為n1與n2[]如果輸入的任一整數小於等於0,或者非數字則輸出**Error!**),其最小公倍數是同時為n1與n2的倍數中最小者(意即可以同時被n1與n2整除的數字中最小者)。

2025/11/19 11:22 Turnin作業7

此題執行結果可參考如下:

[3:23 user@ws hw] ./a.out↩ Please₄input₄two₄numbers(a,b):₄12₄13↔ LCM(12,13)_▲=_▲156← [3:23 user@ws hw] ./a.out← Please₄input₄two₄numbers(a,b): ▲12 ▲5↔ LCM(12,5)_{▲=} ▲60← [3:23 user@ws hw] ./a.out↩ Please₄input₄two₄numbers(a,b): ▲2 ▲ 2 ← LCM(2,2)_{▲=} ▲2← [3:23 user@ws hw] ./a.out← Please₄input₄two₄numbers(a,b):₄0₄2↔ [3:23 user@ws hw] ./a.out↩ Please₄input₄two₄numbers(a,b): 4 - 1 45 ← Error!← [3:23 user@ws hw] ./a.out↩ Please₄input₄two₄numbers(a,b):₄1₄a↔ Error!← [3:23 user@ws hw]

P2 最大公因數□GCD.c□

兩個數字的最大公因數是能夠同時被兩數整除的最大正整數。請設計一個C語言的程式GCD.c□讓使用者輸入兩個不超過10000的正整數(如果不符合或是輸入非數字時則請輸出Error!),請找出這兩個數字的最大公因數□Greatest Common Divisor□GCD□後加以輸出。

本題的輸出結果可以參考以下的內容:

```
[3:23 user@ws hw] ./a.out
PleaseAinputAtwoAnumbers(a,b):A18A4
GCD(18,A4)A=A2
[3:23 user@ws hw] ./a.out
PleaseAinputAtwoAnumbers(a,b):A256A40
GCD(256,A40)A=A8
[3:23 user@ws hw] ./a.out
PleaseAinputAtwoAnumbers(a,b):A80A80
GCD(80,A80)A=A80
[3:23 user@ws hw] ./a.out
PleaseAinputAtwoAnumbers(a,b):A1A1
PleaseAinputAtwoAnumbers(a,b):A-1A1
Error!
```

Last update: 2025/11/18 18:19

```
[3:23 user@ws hw] ./a.out.
PleaseAinputAtwoAnumbers(a,b):A1A0.

Error!.
[3:23 user@ws hw] ./a.out.
PleaseAinputAtwoAnumbers(a,b):A10001A1.

Error!.
[3:23 user@ws hw] ./a.out.
PleaseAinputAtwoAnumbers(a,b):AaA4.

Error!.
[3:23 user@ws hw] ./a.out.

[3:23 user@ws hw] ./a.out.

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

P3 輸出菱形□PrintDiamond.c□

請設計一個C語言程式PrintDiamond.c□讓使用者輸入為一個大於等於3的整數 \$n\$□使用迴圈以及加號印出邊長等於 \$n\$ 的菱形。

如果輸入不符合 \$n >= 3\$ 或是輸入非數字則輸出Error!□

其執行結果可參考如下:

【注意:為了便利閱讀,在以下的輸出結果裡使用了等寬字型顯示題出所輸出的+與。儘管使用了灰色底色但它們仍為輸出結果。】

【注意:本題會採用人工批改,請務必使用迴圈配合printf("+")與printf("")來印出加號與空白,不可以直接使用printf()印出整行的結果。】

```
[3:23 user@ws hw] ./a.out↩
▲▲▲▲+←
▲ ▲ ▲ +++←
▲ ▲ + + + + + ←
 ▲+++++
+++++++
▲+++++
 ▲ ▲ + + + + + ←
▲ ▲ ▲ +++←
A A A A +
[3:23 user@ws hw] ./a.out↩
Please input an ainteger agreater at han a or a equal ato a 3: a 3 d
▲▲+←
 ▲+++
+++++
▲+++←
▲▲+←
[3:23 user@ws hw]./a.out↔
Please input an ainteger agreater at han a or a equal a to a 3: a 1 a
```

2025/11/19 11:22 5/13 Turnin作業7

Error!←

[3:23 user@ws hw] ./a.out↩

PleaseAinputAanAintegerAgreaterAthanAorAegualAtoA3:Age

Error! ←

[3:23 user@ws hw]

P4 計算1到n的質數之和□PrimeNumSum.c□

請設計一個C語言程式PrimeNumSum.c□讓使用者輸入一個正整數 \$n\$□將 1到 \$n\$ 之間的所有質數輸出並加總,如果輸入負數或是非數字則輸出**Error!**,其執行結果可參考如下:

[3:23 user@ws hw] ./a.out↩

Please₄input₄an₄integer: 40↔

2a+a3a+a5a+a7a+a11a+a13a+a17a+a19a+a23a+a29a+a31a+a37a=a197

[3:23 user@ws hw] ./a.out4

Please₄input₄an₄integer: ▲1←

There₄isn't₄any₄prime₄numbers₄between₄1₄and₄1.↩

[3:23 user@ws hw] ./a.out↩

Please₄input₄an₄integer:₄g←

Error! ←

[3:23 user@ws hw] ./a.out↔

Please₄input₄an₄integer: ₄ - 1 ↔

Error!←

[3:23 user@ws hw]

P5 階乘∏Factorial.c□

一個正整數的階乘,表示從1到該數之間所有正整數的乘積。例如,5的階乘記作\$5!\$,計算方法如下[]\$\$5!=1\times2\times3\times4\times5\$\$此外,根據數學定義,\$0!\$的值被定義為\$1\$。

請設計一個C語言的程式Factorial.c□根據使用者輸入的整數,將其階乘的值輸出,其輸出結果可以參考以下的內容:

【注意:本題的輸入不會是大於19的非負整數 , 如果不符合定義或是輸入非數字則輸出Error!□□

【注意:本題會採用人工批改,請務必使用迴圈計算n的階乘後輸出,而非直接印出1~19的階乘值。】

[3:23 user@ws hw] ./a.outa PleaseAinputAanAinteger:A0a

Jun Wu的教學網頁 國立屏東大學資訊工程學系 CSIE. NPTU

```
0!▲=▲1←
[3:23 user@ws hw] ./a.out←
Please₄input₄an₄integer: ▲1←
1!▲=▲1←
[3:23 user@ws hw] ./a.out←
Please₄input₄an₄integer: ₄7↔
7!a=a1a*a2a*a3a*a4a*a5a*a6a*a7a=a5040
[3:23 user@ws hw] ./a.out↩
Please₄input₄an₄integer: ▲20←
Error! ←
[3:23 user@ws hw] ./a.out←
Error!←
[3:23 user@ws hw] ./a.out↩
Please₄input₄an₄integer:₄a↔
Error!←
[3:23 user@ws hw]
```

P6 中空倒三角形□HollowInvertedTriangle.c□

題目敘述:

請設計一個 C□HollowInvertedTriangle.c□語言程式,要求如下:

- 使用者輸入一個整數 n□3 ≤ n ≤ 99□□如果輸入不在這個範圍的整數或是輸入非數字,則輸出Error!□
- 程式輸出由星號(*)組成的倒三角形圖案;
- 圖案共 n 列,其中第一列印出 n 個 *,最後一列印出單一個 *;
- 第二列到第 n-1 列為「中空」倒三角形,每列僅最左與最右印出*,中間填空白;

輸出結果請參考如下:

【注意:為了便利閱讀,在以下的輸出結果裡使用了等寬字型顯示題出所輸出的+與。儘管使用了灰色底色但它們仍為輸出結果。】

【注意:本題會採用人工批改,請務必使用迴圈配合printf("*")與printf("")來印出加號與空白,不可以直接使用printf()印出整行的結果。】

```
[3:23 user@ws hw] ./a.out--
5--

*****--

***--

**--

**--

**--

[3:23 user@ws hw] ./a.out--
9--
```

2025/11/19 11:22 7/13 Turnin作業7

P7 區間乘法表□RangeMultiplicationTable.c□

題目敘述:

請設計一個 C 語言程式□RangeMultiplicationTable.c□□讓使用者輸入 兩個整數 a 與 b□不分大小,例如 a=9□b=11 或 a=11□b=9 都可)。

規則說明:

假設輸入兩個整數 a 與 b □程式需先判斷哪一個為較小值、哪一個為較大值,然後印出 區間內(含端點) 所有數字的乘法表。

如果輸入內容超過兩個整數或是輸入非數字,則輸出Error!←[每一筆輸出格式為□\${x}\times{y}=結果\$

每個 x 的乘法表需包含以下內容:

- y 從較小值到較大值依序列印
- 完成一個 x 的乘法表後,要換行再印下一個 x 的乘法表

```
[3:23 user@ws hw]./a.out~
Please₄input₄two₄integers: ▲1 ▲ 5 ↔
1<sub>A</sub>×<sub>A</sub>1<sub>A</sub>=<sub>A</sub>1←
1<sub>▲</sub>×<sub>4</sub>2<sub>4</sub>=<sub>4</sub>2←
1<sub>▲</sub>×<sub>▲</sub>3<sub>▲</sub>=<sub>▲</sub>3<sub>←</sub>
1<sub>▲</sub>×<sub>4</sub>4<sub>▲</sub>=<sub>▲</sub>4←
1<sub>A</sub>×<sub>A</sub>5<sub>A</sub>=<sub>A</sub>5←
2▲×▲2▲=▲4←
2▲X▲3▲=▲6←
2▲x▲4▲=▲8←
2<sub>A</sub>×<sub>A</sub>5<sub>A</sub>=<sub>A</sub>10←
3▲x▲3▲=▲9←
3AXA4A=A12←
3<sub>A</sub>×<sub>A</sub>5<sub>A</sub>=<sub>A</sub>15←
4<sub>A</sub>×<sub>A</sub>4<sub>A</sub>=<sub>A</sub>16←
4∆X∆5∆=∆20←
5<sub>A</sub>×<sub>A</sub>5<sub>A</sub>=<sub>A</sub>25←
[3:23 user@ws hw] ./a.out↩
Please₄input₄two₄integers: ▲1 ▲ 3 ↔
1<sub>▲</sub>×<sub>4</sub>1<sub>▲</sub>=<sub>4</sub>1←
1<sub>A</sub>×<sub>A</sub>2<sub>A</sub>=<sub>A</sub>2←
1<sub>A</sub>X<sub>A</sub>3<sub>A</sub>=<sub>A</sub>3←
2<sub>▲</sub>×<sub>▲</sub>2<sub>▲</sub>=<sub>▲</sub>4←
2▲X▲3▲=▲6←
3▲x▲3▲=▲9←
[3:23 user@ws hw] ./a.out↩
Please input atwo integers: 49 15 ←
9<sub>A</sub>×<sub>A</sub>9<sub>A</sub>=<sub>A</sub>81←
9<sub>A</sub>×<sub>A</sub>10<sub>A</sub>=<sub>A</sub>90←
9<sub>A</sub>×<sub>A</sub>11<sub>A</sub>=<sub>A</sub>99←
9<sub>A</sub>×<sub>A</sub>12<sub>A</sub>=<sub>A</sub>108←
9<sub>A</sub>×<sub>A</sub>13<sub>A</sub>=<sub>A</sub>117←
9<sub>A</sub>×<sub>A</sub>14<sub>A</sub>=<sub>A</sub>126←
9<sub>A</sub>x<sub>A</sub>15<sub>A</sub>=<sub>A</sub>135←
10<sub>A</sub>×<sub>A</sub>10<sub>A</sub>=<sub>A</sub>100←
10<sub>A</sub>×<sub>A</sub>11<sub>A</sub>=<sub>A</sub>110←
10<sub>A</sub>×<sub>A</sub>12<sub>A</sub>=<sub>A</sub>120←
10<sub>A</sub>×<sub>A</sub>13<sub>A</sub>=<sub>A</sub>130←
10<sub>A</sub>×<sub>A</sub>14<sub>A</sub>=<sub>A</sub>140←
10<sub>A</sub>×<sub>A</sub>15<sub>A</sub>=<sub>A</sub>150←
11<sub>▲</sub>×<sub>▲</sub>11<sub>▲</sub>=<sub>▲</sub>121←
```

2025/11/19 11:22 9/13 Turnin作業7

```
11<sub>A</sub>×<sub>A</sub>12<sub>A</sub>=<sub>A</sub>132←
11▲X▲13▲=▲143←
11<sub>A</sub>×<sub>A</sub>14<sub>A</sub>=<sub>A</sub>154←
11<sub>A</sub>×<sub>A</sub>15<sub>A</sub>=<sub>A</sub>165←
12▲X▲12▲=▲144←
12▲X▲13▲=▲156←
12<sub>A</sub>×<sub>A</sub>14<sub>A</sub>=<sub>A</sub>168←
12<sub>A</sub>×<sub>A</sub>15<sub>A</sub>=<sub>A</sub>180←
13<sub>A</sub>×<sub>A</sub>13<sub>A</sub>=<sub>A</sub>169←
13<sub>▲</sub>×<sub>▲</sub>14<sub>▲</sub>=<sub>▲</sub>182←
13<sub>A</sub>×<sub>A</sub>15<sub>A</sub>=<sub>A</sub>195←
14<sub>▲</sub>×<sub>▲</sub>14<sub>▲</sub>=<sub>▲</sub>196←
14<sub>A</sub>x<sub>A</sub>15<sub>A</sub>=<sub>A</sub>210←
15<sub>▲</sub>x<sub>▲</sub>15<sub>▲</sub>=<sub>▲</sub>225←
[3:23 user@ws hw] ./a.out↩
Please₄input₄two₄integers: 410 48 ↔
8<sub>▲</sub>×<sub>▲</sub>8<sub>▲</sub>=<sub>▲</sub>64←
8<sub>A</sub>×<sub>A</sub>9<sub>A</sub>=<sub>A</sub>72←
8<sub>A</sub>×<sub>A</sub>10<sub>A</sub>=<sub>A</sub>80←
9<sub>A</sub>×<sub>A</sub>9<sub>A</sub>=<sub>A</sub>81←
9<sub>A</sub>×<sub>A</sub>10<sub>A</sub>=<sub>A</sub>90←
10<sub>A</sub>×<sub>A</sub>10<sub>A</sub>=<sub>A</sub>100←
[3:23 user@ws hw] ./a.out↩
Please input atwo integers: a3 a - 1 ←
-1▲X▲-1▲=▲1←
-1AXA0A=A0←
-1▲X▲1▲=▲-1←
-1<sub>A</sub>×<sub>A</sub>2<sub>A</sub>=<sub>A</sub>-2←
-1▲X▲3▲=▲-3←
0▲×▲0▲=▲0←
0<sub>A</sub>×<sub>A</sub>1<sub>A</sub>=<sub>A</sub>0←
0<sub>▲</sub>×<sub>4</sub>2<sub>▲</sub>=<sub>4</sub>0←
0▲×▲3▲=▲0←
1<sub>▲</sub>×<sub>▲</sub>1<sub>▲</sub>=<sub>▲</sub>1←
1<sub>▲</sub>×<sub>▲</sub>2<sub>▲</sub>=<sub>▲</sub>2←
1<sub>▲</sub>×<sub>▲</sub>3<sub>▲</sub>=<sub>▲</sub>3<sub>←</sub>
2▲×▲2▲=▲4←
2<sub>A</sub>x<sub>A</sub>3<sub>A</sub>=<sub>A</sub>6←
3▲X▲3▲=▲9←
```

Jun Wu的教學網頁 國立屏東大學資訊工程學系 CSIE, NPTU

Last update: 2025/11/18 18:19

```
[3:23 user@ws hw]./a.out~
Please₄input₄two₄integers:₄-2₄3↔
-2AXA-2A=A4←
-2AXA-1A=A2←
-2▲X▲0▲=▲0←
-2▲X▲1▲=▲-2←
-2▲×▲2▲=▲-4←
-2AXA3A=A-6←
-1AXA-1A=A1←
-1▲X▲0▲=▲0←
-1<sub>A</sub>X<sub>A</sub>1<sub>A</sub>=<sub>A</sub>-1←
-1<sub>A</sub>×<sub>A</sub>2<sub>A</sub>=<sub>A</sub>-2←
-1<sub>A</sub>X<sub>A</sub>3<sub>A</sub>=<sub>A</sub>-3←
0▲×▲0▲×▲0
0<sub>A</sub>×<sub>A</sub>1<sub>A</sub>=<sub>A</sub>0←
0<sub>▲</sub>×<sub>▲</sub>2<sub>▲</sub>=<sub>▲</sub>0←
0▲×▲3▲=▲0←
1<sub>▲</sub>×<sub>▲</sub>1<sub>▲</sub>=<sub>▲</sub>1←
1<sub>A</sub>×<sub>A</sub>2<sub>A</sub>=<sub>A</sub>2←
1<sub>A</sub>x<sub>A</sub>3<sub>A</sub>=<sub>A</sub>3←
2<sub>▲</sub>×<sub>▲</sub>2<sub>▲</sub>=<sub>▲</sub>4←
2▲X▲3▲=▲6←
3▲X▲3▲=▲9€
[3:23 user@ws hw] ./a.out↔
Please input two integers: Aa A 1 4
Error!←
[3:23 user@ws hw]
```

P8 數字回文金字塔□NumPalindromePyramid.c□

請設計一個C語言的程式NumPalindromePyramid.c□讓使用者輸入一個正奇數 \$n\$ 以產生一個數字回文金字塔,假設輸入的數字為\$n\$□如果\$n <= 0\$ 或是輸入非數字則輸出**Error!**□

輸出的數字回文金字塔的高度(層數)為\$\lceil \frac{n}{2} \rceil\$□且每層可參考下列的執行結果輸出數字回文:

【注意:為了便利閱讀,在以下的輸出結果裡使用了等寬字型顯示題出所輸出的+與。儘管使用了灰色底色但它們仍為輸出結果。】

【注意:本題會採用人工批改,請務必使用迴圈配合printf()印出|、數字與空白,不可以直接使用printf()印出整行的結果。】

```
[3:23 user@ws hw] ./a.out~
```

2025/11/19 11:22 Turnin作業7

[3:23 user@ws hw] ./a.out↩

Pleaseainputatheanumberaofathealayeraofatheapyramid: \$\alpha = \frac{1}{2} \text{ Pleaseainputatheanumberaofathealayeraofatheapyramid: }\alpha = \frac{1}{2} \text{ Pleaseainputatheanumberaofathealayeraofatheapyramid: }\alpha = \frac{1}{2} \text{ Pleaseainputatheanumberaofathealayeraofatheapyramid: }\alpha = \frac{1}{2} \text{ Pleaseainputatheanumberaofatheapyramid: }\alpha = \frac{1}{2} \text{ Pleaseainputatheanumberaofatheapyramid: }\alpha = \frac{1}{2} \text{ Pleaseainputatheapyramid: }\alpha = \fra

Please input the anumber of the alayer of the apyramid: 494

| A A 1 A A | 44 | A 121 A | 44 | 12321 | 44

[3:23 user@ws hw] ./a.out↩

Pleaseainputatheanumberaofathealayeraofatheapyramid: 14

|1|↩

[3:23 user@ws hw]./a.out↩

Pleaseainputatheanumberaofathealayeraofatheapyramid: 44

Error!←

[3:23 user@ws hw] ./a.out←

Please input the anumber of the alayer of the apyramid: 1-1-

Error!←

[3:23 user@ws hw] ./a.out↩

Please input the anumber of the alayer of the apyramid: 404

Error! ←

[3:23 user@ws hw] ./a.out↩

Please input the anumber of the alayer of the apyramid: Aad

Error!←

[3:23 user@ws hw]

P9 數字金字塔□NumberPyramidWithAlignment.c□

題目敘述:請設計一個 C 語言程式,要求如下:

使用者輸入一個整數 $n \mid 1 \le n \le 9 \mid \mid 1 \le n \le 9$ 世初果輸入不在這個範圍的數值或是輸入非數字則輸出**Error!** 接著輸入一個字元 $n \mid 1 \le n \le 9$ 世初果輸入不在這個範圍的數值或是輸入非數字則輸出**Error!** 日本的表面,

- 如果輸入非 I, c, r 的英文字元, 則輸出Error!
- Ⅰ表示 靠左輸出;
- c表示 靠中輸出,每列數字從左到右依序排列,並在每列前加適當空格。使圖形整體水平居中,如果使用靠中輸出,則必須輸入奇數,否則輸出Error!□
- r表示靠右輸出,每列數字從左到右依序排列,並在每列前加適當空格,使整體右對齊;

Jun Wu的教學網頁 國立屏東大學資訊工程學系 CSIE. NPTU

請印出數字金字塔圖案:

- 每列從 1 開始累加到該列的列數;
- 根據對齊設定調整每列前的空格數量;
- 每列印完後換行;

執行結果請參考如下:

【注意:為了便利閱讀,在以下的輸出結果裡使用了等寬字型顯示題出所輸出的+與 。儘管使用了灰色底色但它們仍為輸出結果。】

【注意:本題會採用人工批改,請務必使用迴圈配合printf()印出數字與空白,不可以直接使用printf()印出整行的結果。】

```
[3:23 user@ws hw] ./a.out↩
▲▲▲▲▲▲▲
▲▲▲▲▲1▲2←
▲▲▲▲1▲2▲3←
▲1▲2▲3▲4←
1▲2▲3▲4▲5←
[3:23 user@ws hw]./a.out↔
Please input a integer and a a char: 46 1 4
1←
1 ▲ 2 ←
1 ▲ 2 ▲ 3 ←
1 ▲ 2 ▲ 3 ▲ 4 ←
1 ▲ 2 ▲ 3 ▲ 4 ▲ 5 ←
1 ▲ 2 ▲ 3 ▲ 4 ▲ 5 ▲ 6 ←
[3:23 user@ws hw] ./a.out↩
Please input a a integer and a a char: 49 A C ~
▲▲▲▲▲▲▲1←
▲▲▲▲▲▲1▲2←
▲▲▲▲▲1▲2▲3←
▲▲▲▲1▲2▲3▲4←
▲▲▲▲1▲2▲3▲4▲5←
AAA1A2A3A4A5A6←
▲1▲2▲3▲4▲5▲6▲7←
▲1▲2▲3▲4▲5▲6▲7▲8←
1<sub>4</sub>2<sub>4</sub>3<sub>4</sub>4<sub>5</sub>46<sub>4</sub>7<sub>4</sub>8<sub>4</sub>9<sub>4</sub>
[3:23 user@ws hw] ./a.out←
Please input a integer and a a char: 12 14
1←
1▲2←
[3:23 user@ws hw],/a.out4
Please input a a integer and a a char: A 2 A r ~
▲▲1←
1▲2←
[3:23 user@ws hw] ./a.out↩
Please input a a integer a and a a char: A 2 A C ~
Error!←
[3:23 user@ws hw] ./a.out↩
```

2025/11/19 11:22 13/13 Turnin作業7

Please input a integer and a a char: 40 A C ~

Error!←

[3:23 user@ws hw]./a.out₄

Please input a a integer a and a a char: A 3 A q ~

Error!←

[3:23 user@ws hw] ./a.out↩

Please input a a integer a and a a char: 1 2 3 4

Error!←

[3:23 user@ws hw]

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: 237076

Permanent link:

https://junwu.nptu.edu.tw/dokuwiki/doku.php?id=c:2025fall-hw7

Last update: 2025/11/18 18:19

