

National Sun Yat-sen University Department of Computer Science and Engineering
C Programming Mid-Term

Note :

- (1) 程式碼全部撰寫至同一 .c/.cpp 檔中
- (2) 每一題輸出結果輸出皆須標明題號並空一行來區隔
- (3) 每一題要能判斷輸入的範圍
- (4) .c/.cpp、資料夾以及壓縮檔檔名請用自己的學號命名

1. 請印出自己的學號、姓名和系別。(5%)

ex : B993040100 王阿明 資訊工程學系

2. 讓使用者輸入一個數字 $N(-50 \leq N \leq 50)$ ，與自己的學號後五碼進行加減乘除 (+ - * /) 運算，並列印出來。(10%)

Note : (I) N 變數型態為 double。

(II) 需對 N 的範圍進行判斷，如果不在範圍內則印出錯誤訊息。

(III) 每一筆輸出結果需換行。

ex :

Input : -50.0

Output : 學號後五碼 : 40100

+ : $40100 + (-50.0) = 40050.0$

- : $40100 - (-50.0) = 40150.0$

* : $40100 * (-50.0) = -2005000.0$

/ : $40100 / (-50.0) = -802.0$

Input : 25.2

Output : 學號後五碼 : 40100

+ : $40100 + (25.2) = 40025.2$

- : $40100 - (25.2) = 40074.8$

* : $40100 * (25.2) = 1010520.0$

/ : $40100 / (25.2) = 1591.269$

Input : 80.0

Output : 學號後五碼 : 40100

N out of range!

3. 請將學號後五碼反轉印出，並撰寫一個附函式(格式如下)來比較反轉前(ori)、反轉後(rev)以及第二題所輸入值(N)的大小，並印出其最大值。(15%)

Note : (I)double Maximum(double ori, double rev, double N); 。

(II)反轉時不允許直接印出，需使用計算方法來達成。

(III)每一筆輸出結果需換行。

ex :

假設 N = 25.2

學號後五碼：40100

Onput :

Reverse : 104

Maximum = 40100.0

4. 請將第二題所輸入的 N 值，若 $N < 0$ 則轉換為正數，再使用 while 迴圈求出 $\leq N$ 值的最大整數 M 並印出。(5%)

ex :

假設 N = 25.2

Onput :

M = 25

假設 N = -15.6

Onput :

M = 15

5. 請先讓使用者輸入一個字元 C (A~H)，並將第四題求出的 M 值，乘上字元 C 在英文字母的序號(A->1、B->2...以此類推)，得到一整數 R。接著使用 for 迴圈判斷 0~R 之間可以被 4、7、11 所整除的數字分別印出。(15%)

Note : (I)字元 C 轉成序號請使用 switch 來判斷。

ex :

假設 M = 15

Input : C

Output :

R = 45

4 : 4 8 12 16 20 24 28 32 36 40 44

7 : 7 14 21 28 35 42

6. 請將第五題所產生的 R 值當成直徑，算出以此 R 為直徑的圓形面積、周長以及以此 R 為直徑的正球體體積及表面積，並印出。(10%)

Note : (I)圓周率 $PI = 3.141592$

(II)圓周長 = $2 * PI * r$ (r 為半徑).

(III)圓面積 = $PI * r^2$ (r 為半徑).

(III)正球體體積 = $4/3 * PI * r^3$ (r 為半徑).

(IV)正球體表面積 = $4 * PI * r^2$ (r 為半徑).

ex :

假設 $R = 45$

Output :

圓周長 = 141.371641

圓面積 = 1590.430963

正球體體積 = 47712.928891

正球體表面積 = 6361.723852

7. 讓使用者輸入兩正整數 $Z1(0 < Z1 \leq 100)$ 以及 $Z2$ ，使用 for 迴圈求出 $\leq Z1$ 的最大質數 P，接著將 P 和 $Z2$ 間的所有整數(包含 P 以及 $Z2$)由大至小印出並且求其總合、平均。(10%)

Note : (I)每一筆輸出結果需換行。

ex :

Input : 15 -5

Output :

Max Prime = 13

13 12 11 10 9 8 7 6 5 4 3 2 1 0 -1 -2 -3 -4 -5

Sum = 76

Average = 4.0

8. 將上題所求到的 P，印出 P 除以 10 所得到的餘數(r)，並計算印出 0 到 r 的階乘值。(10%)

Note : (I)每一筆階乘輸出需換行。

(II) $n! = n*(n-1)*(n-2)*\dots*1$ 。

ex :

假設 Max Prime = 19

Output :

r = 9

0 ! = 1
1 ! = 1
2 ! = 2
3 ! = 6
4 ! = 24
5 ! = 120
6 ! = 720
7 ! = 5040
8 ! = 40320
9 ! = 362880

9. **Palindrome(迴文)**，指的是一個數字以由左至右以及由右至左的方式來讀其值皆為相同的(Ex : 9339, 252, 7667...), 而我們所要撰寫的程式就為輸入一個數字不斷的與自己的 Reverse(反數)進行相加，就一直重複這個程序，直到我們的結果出現 Palindrome 為止，並把流程列印出來。(20%)

Note : (I)流程進行以 10 次為止，計算 10 次之後還未出現迴文則中斷跳出。

ex :

Input : 195

195 //開始的數字

591

786

687

1473

3741

5214

4125

9339 //進行四次之後，**Palindrome** 出現了

National Sun Yat-sen University Department of Computer Science and Engineering
C Programming Mid-Term 2

- (1) 程式碼全部撰寫至同一 .c/.cpp 檔中
- (2) 每一題輸出結果輸出皆須標明題號並空一行來區隔
- (3) .c/.cpp、資料夾以及壓縮檔檔名請用自己的學號命名

1.1 請使用者輸入兩正整數 N、M，N 代表水平線單位長度(代表印出-數量)，M 代表線段單位長度次數，‘*’ 代表中央原點(0,0)、+代表由原點算起每 N 倍數單位時符號，請參照以下範例畫出相對應水平線(5%)。

Note :

- (1) 函式型態：void horizontal_Line(int N, int M);
- (2) 線段開始及結束需多印兩‘-’ 符號，並且結束須用多印箭頭‘>’ 及‘x’ 符號

Example :

Input : 10 2

Output :

```
--+-----+-----*-----+-----+--> x
```

Input : 8 1

Output :

```
--+-----*-----+--> x
```

1.2 讓使用者輸入兩字元符號分別取代‘-’ 以及‘+’ 符號，並印出其結果(5%)。

Note :

- (1) 函式型態：void horizontal_Line_Char(int N, int M, char a, char b);

Example :

M = 10, N = 2

Input : =!

Output :

```
--!=====!=====*=====!=====!--> x
```

M = 8, N = 1

Input : ~?

Output :

```
--?~~~~~*~~~~~?--> x
```

2.1 延續第 1 題，保留上題的 x 軸，請使用者輸入兩正整數 O、P，O 代表垂直線單位長度(代表印出|數量)，P 代表線段單位長度次數，‘*’ 代表中央原點(0,0)、+ 代表由原點算起每 O 倍數單位時符號，請參照以下範例往下畫出相對應垂直線(7%)。

Note :

(1)函式型態：void vertical_Line(int N, int M, int O, int P);

(2)線段結束需多印一‘|’ 符號

Example :

M = 10, N = 2

Input : 3 2

Output :

```

--+-----+-----*-----+-----+--> x
    |
    |
    +
    |
    |
    +
    |

```

2.2 讓使用者輸入三字符號分別取代‘-’、‘|’ 以及‘+’ 符號，並印出其結果(8%)。

Note :

(1)函式型態：void vertical_Line_Char(int N, int M, int O, int P, char c, char d, char e);

Example :

M = 10, N = 2, O = 3, P = 2

Input : ^ ~ \$

Output :

```

--$^^^^^^^^^$^^^^^^^^^*^^^^^^^^^$^^^^^^^^^$--> x
    ~
    ~
    $
    ~
    ~
    $
    |

```

- 延續第 2 題，保留上題的 x、y 軸，延續上題的 O 與 P 值，O 代表垂直線單位長度(代表印出|數量)，P 代表線段單位長度次數，'*' 代表中央原點(0,0)、'+' 代表由原點算起每 O 倍數單位時符號，請參照以下範例往上畫出相對應垂直線(10%)。

Note :

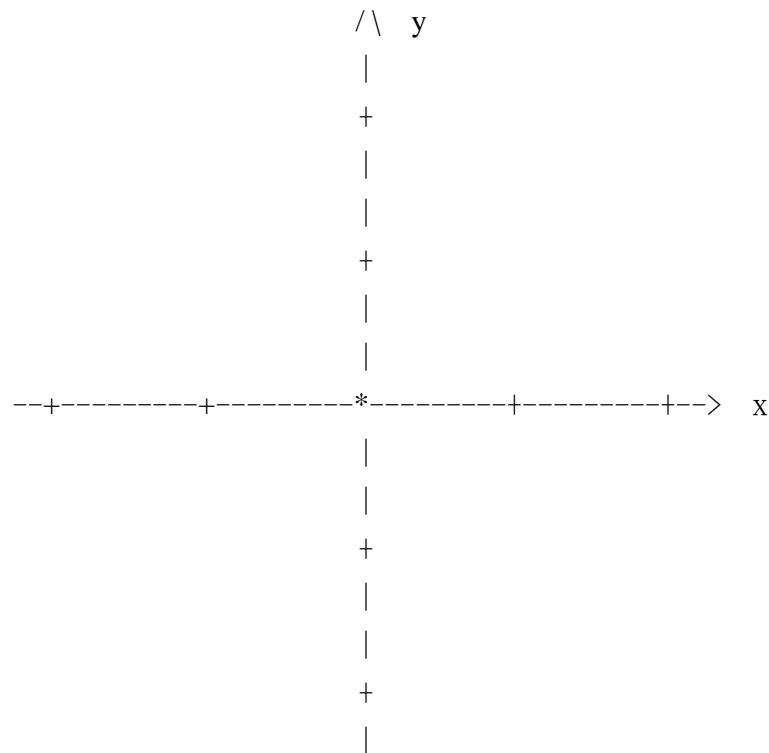
(1)函式型態：void cross_Line(int N, int M, int O, int P);

(2)線段開始及結束需多印一'|' 符號，並且結束須用多印箭頭'\>' 以及 'y' 符號。

Example :

M = 10, N = 2, O = 3, P = 2

Output :



- 延續上題，讓使用者輸入兩整數與一字元，兩整數 x、y 以及一字元 f，每個符號代表一單位，由原點到點(x,y)區域皆用字元 f 填滿印出(15%)。

Note :

(1) 函式型態：void cross_part (int N, int M, int O, int P , int x, int y, char f);

(2) 判斷輸入的(x,y)是否在該單位內，若超過將該單位全填滿。

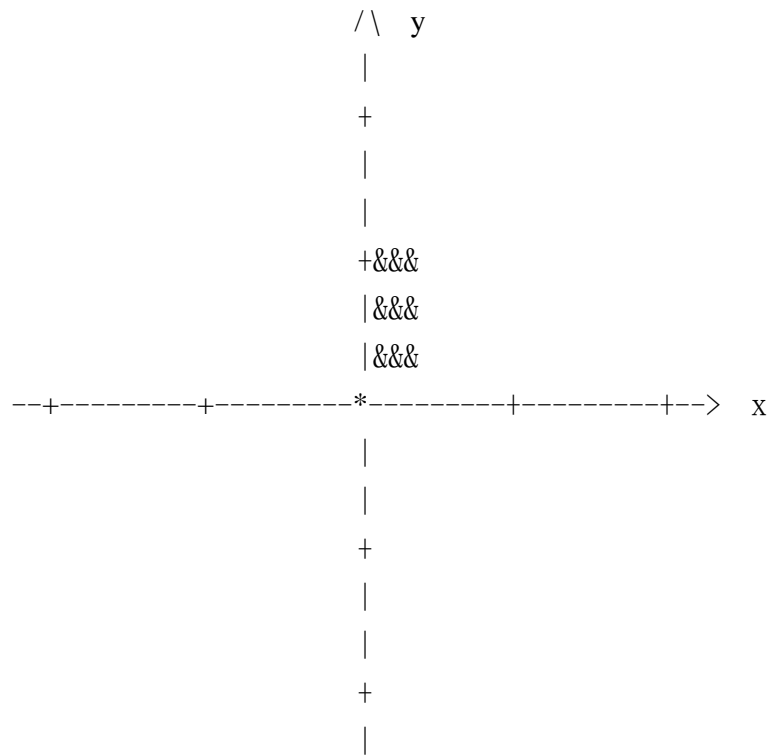
(3) 若是(x,y)在 x 軸或 y 軸上，保持原圖。

Example :

M = 10, N = 2, O = 3, P = 2

Input : 3 3 &

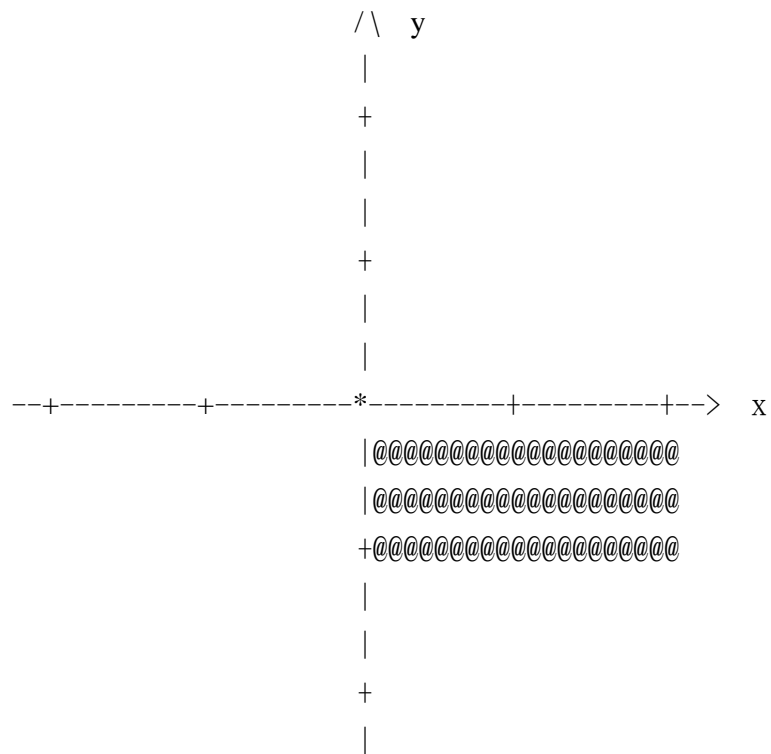
Output :



M = 10, N = 2, O = 3, P = 2

Input : 25 -3 @

Output :



5. 延續第 3 題，讓使用者輸入一字元，用此字元畫出 $y=x$ 線段(15%)。

Note :

(1) 函式型態：`void yx_Line(int N, int M, int O, int P, char g);`

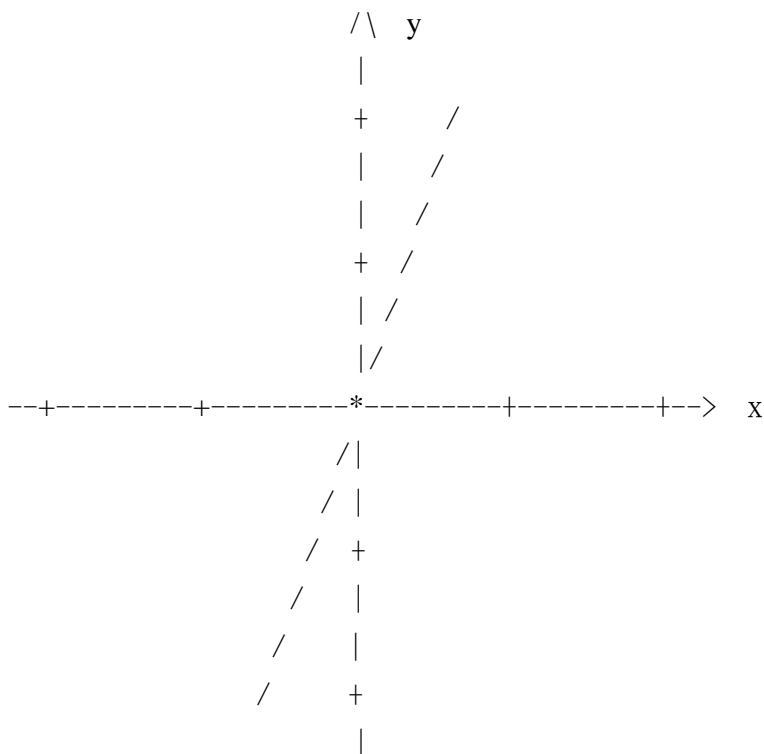
(2) 每個符號代表一單位。

Example :

M = 10, N = 2, O = 3, P = 2

Input : /

Output :



6. 讓使用者輸入一個數字 R, 代表 $0 \sim \pi$ 間(不包含原點)畫出的點數, y 軸固定長度 10, 請畫出 $\cos(x)$ 的圖形(15%)。

Note :

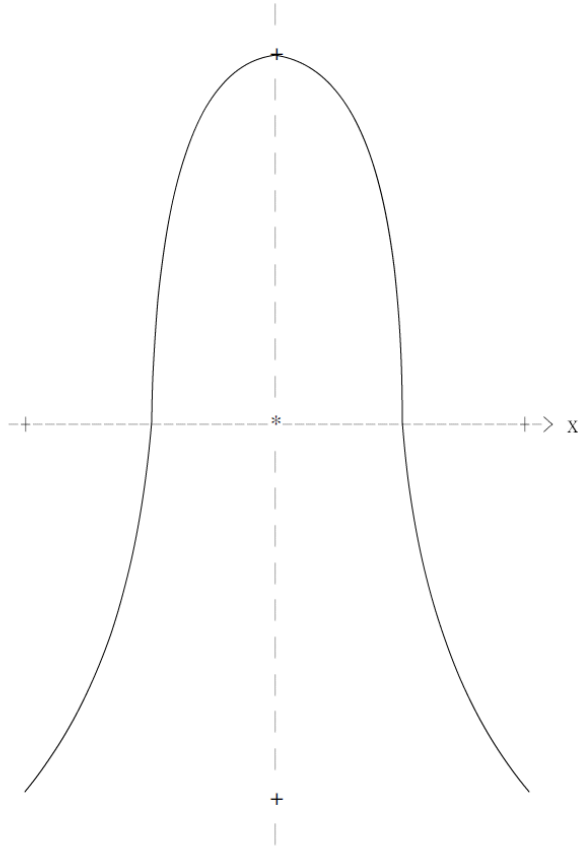
(1) 函式型態：`void cos_graph(int R);`

(2) 每個符號代表一單位。

Example :

Input : 20

Output :



7.將以上六題的結果，輸出至 output.txt(10%)。

Note :

(1)

函式型態：void Fhorizontal_Line(int N, int M, FILE *fp);

void Fhorizontal_Line_Char(int N, int M, char a, char b, FILE *fp);

void Fvertical_Line(int N, int M, int O, int P, FILE *fp);

void Fvertical_Line_Char(int N, int M, int O, int P, char c, char d, char e,
FILE *fp);

void Fcross_Line(int N, int M, int O, int P, FILE *fp);

void Fcross_part (int N, int M, int O, int P, int x, int y, char f, FILE *fp);

void Fyx_Line(int N, int M, int O, int P, char g, FILE *fp);

void Fcos_graph(int R, FILE *fp);

(2)每題結果請標明題號並換行區隔

8. 請撰寫一個程式，畫出下面的圖形(10%)。

Note :

(1) 改變顏色 function，請用(詳情請看附的資料)：

```
BOOL SetConsoleTextAttribute( HANDLE hConsoleOutput,
WORD wAttributes);
```

(2)請用 for 迴圈，不可直接印出，只能使用 printf(“*”)、printf(“\n”)。

(3)樹的部分有 15 層，根有 2 層，盆栽有 5 層。

