



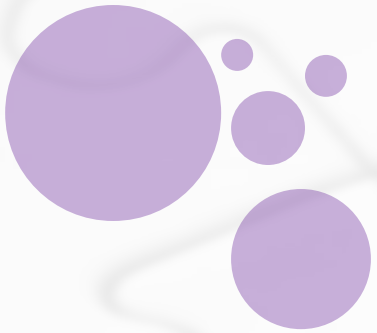
2018

程式設計
沒有加強班

程式設計與實習(二)

BY 孫茂勛

Email:JOHN85051232@GMAIL.COM



一個程式語言的基本概念

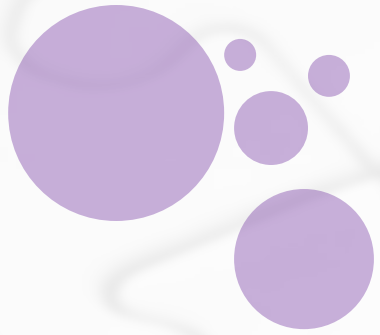
- 開專案
- Hello World
- 資料型態(Data type)
- 變數(Variable)
- 輸入輸出(Input/Output)
- 流程控制(If...else...)
- 迴圈(Loop)
- 函式(Function)
- 陣列(Array)
- 指標(Pointer)
- 字串(String)
- 讀寫檔(File I/O)



一個程式語言的基本概念

- 🔵 恭喜，到這裡其實你已經可以說你算學完C語言了
- 🔵 這學期要來講一些應用和技巧
- 🔵 如何用學到的東西去寫出更厲害的程式碼

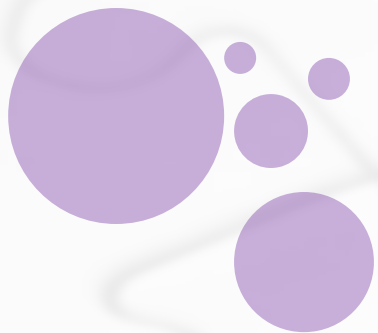




大數運算

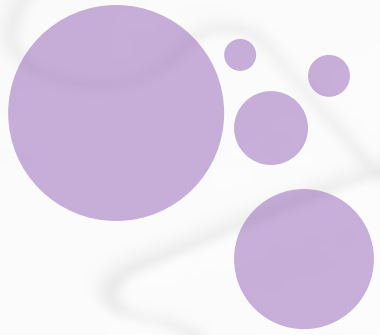
Q : 123456789987654321 + 987654321123456789 = ?





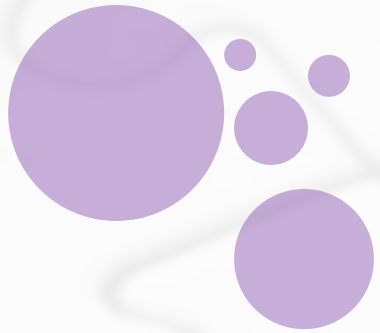
複習一下整數可使用的範圍

<u>int</u>	4bytes	-2147483648 to 2147483647
unsigned <u>int</u>	4bytes	0 to 4294967295
signed int	4bytes	-2147483648 to 2147483647
long <u>int</u>	4bytes	-2,147,483,648 to 2,147,483,647



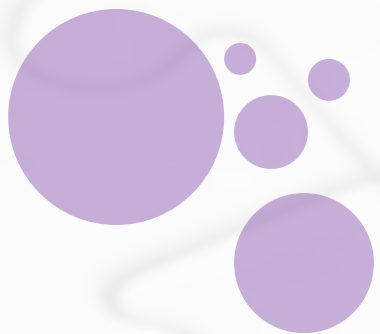
大數運算

- ❖ 用來做很大很大的數的運算
- ❖ 陣列
- ❖ 手算怎麼算的程式就怎麼寫
- ❖ CPE熱門考題(另外一個是質數表)



數字的表示法

1. 陣列
2. ASCII
3. 順序顛倒(可做可不作)



數字的表示法

陣列

ASCII

順序顛倒

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define LEN 100
int main()
{
    char str1[LEN] = {0};
    char str2[LEN] = {0};
    char op = 0;
    while(scanf("%s %c %s",str1,&op,str2)!= EOF)
    {
        printf("%s %c %s = \n",str1,op,str2);
    }
    return 0;
}
```




說到字元又想到ASCII

複習一下

💡電腦只能表示1和0，那要如何表示數字/英文/符號？

💡數字=>轉成二進位，解決！

💡英文和符號？設計一個編碼，用數字代表這些資訊

💡編碼的方式就叫做ASCII

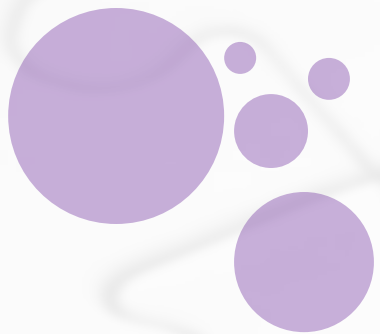
💡<https://zh.wikipedia.org/wiki/ASCII>

說到字元又想到ASCII

9 * 9 = 81

'9' * '9' = ?

	48		0
	49		1
	50		2
	51		3
	52		4
	53		5
	54		6
	55		7
	56		8
	57		9



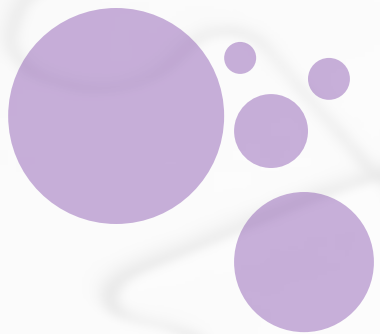
數字的表示法

▣ 陣列

▣ ASCII

- 我們不要 '9' !! 而是要9!!
- How? 減掉 '0'

▣ 順序顛倒




數字的表示法

陣列

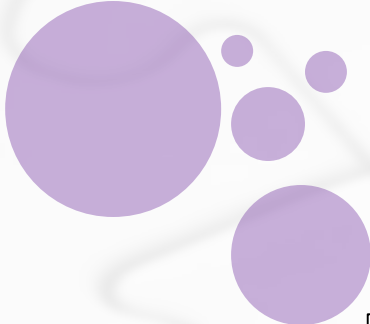
ASCII

順序顛倒

- 原本的最高位數在陣列第0個位置不好操作，WHY?
- 多創一個int陣列把順序顛倒過來



index	0	1	2	3	4
Char[]	1	2	3	4	5
Int[]	5	4	3	2	1



數字的表示法

```
char str1[LEN] = {0};
char str2[LEN] = {0};
int num1[LEN] = {0};
int num2[LEN] = {0};
char op = 0;
while(scanf("%s %c %s",str1,&op,str2)!= EOF)
{
    printf("%s %c %s = \n",str1,op,str2);
    for(int i = 0 ; i < strlen(str1) ; ++i)
    {
        num1[strlen(str1) - 1 - i] = str1[i] - '0';
    }
    for(int i = 0 ; i < strlen(str2) ; ++i)
    {
        num2[strlen(str2) - 1 - i] = str2[i] - '0';
    }
}
```



更新

🔲 每次做完運算記得初始化所有的陣列，以免上次的運算結果還存留在陣列中

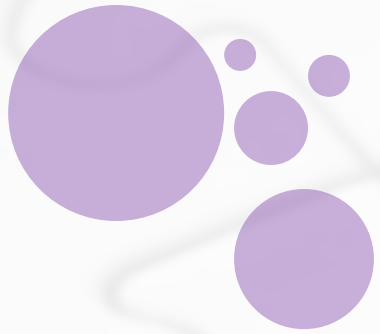
```
for( int i = 0 ; i < LEN ; i++)  
{  
    str1[i] = 0;  
    str2[i] = 0;  
    num1[i] = num2[i] = 0;  
}
```

輸出(更新)

- 前面顛倒過了，所以要在顛倒回來
- 防止有前綴0的情況(Ex:00012345)

```
void print(int num[])
{
    int flag = 0;
    for(int i = LEN*2 - 1 ; i >= 0 ; --i)
    {
        if(num[i] != 0) flag = 1;
        if(flag == 1) printf("%d",num[i]);
    }
    if(flag == 0)printf("0"); //如果答案是0就印出來
    printf("\n");
}
```

```
int main()
{
    ...
    print(num1);
    print(num2);
    ...
}
```



開始講運算之前

🔲 先把運算的判斷寫好

```
switch(op)
{
    case '+':
        add(num1,num2);
        break;
    case '-':
        subtraction(num1,num2);
        break;
    case '*':
        multiple (num1,num2);
        break;
}
```



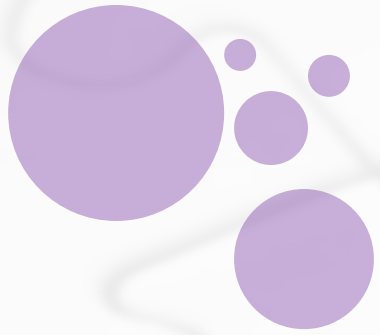

$$\begin{array}{r}
 1111111 \\
 9999999 \\
 + 9999999 \\
 \hline
 1999998
 \end{array}$$

大數運算 - 加法

從最低位元開始算，這樣才能算進位

```
void add(int num1[], int num2[])
{
    int num3[LEN] = {0};
    for(int i = 0 ; i < LEN ; ++i)
    {
        num3[i] += num1[i] + num2[i];
        num3[i+1] += num3[i] / 10;
        num3[i] %= 10 ;
    }
    print(num3);
}
```

	1	1	1	1	1	1	1	
	9	9	9	9	9	9	9	9
	9	9	9	9	9	9	9	9
+	9	9	9	9	9	9	9	8
	1	9	9	9	9	9	9	8



更新

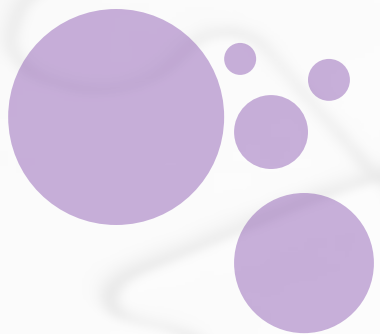
🔲 $10^{10} + 10^{10} = ?$

🔲 最高可能到 10^{11} (進位)

🔲 所以 num3 的陣列大小至少要宣告成 $LEN+1$



Created by free version of DocuFreezer



大數運算 – 減法

❖ 這段程式碼只能在 $\text{num1} > \text{num2}$ 的時候正常執行

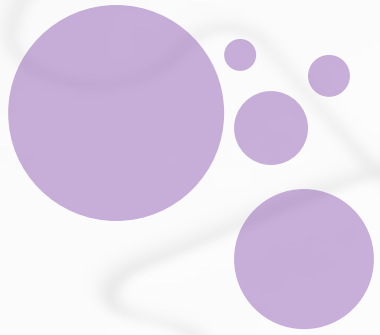
❖ Ex : $1 - 2 = ?$

❖ 小數減大數怎麼寫？

❖ 比較兩數長度和最大次方的係數，找出最大數

❖ 變成大數 – 小數

❖ 輸出的時候自己補上負號



大數運算 – 乘法

$$\begin{array}{r} \text{X} \\ \hline \begin{array}{r} 111 \\ 78 \\ 888 \\ 777 \end{array} \end{array}$$

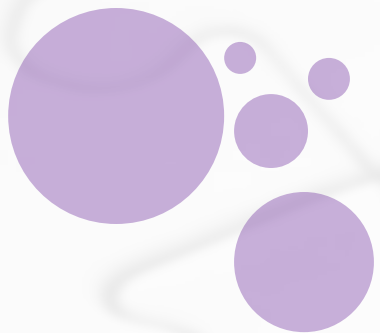
大數運算 – 乘法

```
void multiple(int num1[], int num2[])
{
    int num3[LEN*2] = {0};
    for(int i = 0 ; i < LEN ; ++i)
    {
        if(num1[i] == 0) continue;
        for(int j = 0 ; j < LEN ; ++j)
        {
            num3[i+j] += num1[i] * num2[j];
        }
    }
    for(int i = 0 ; i < LEN*2 ; ++i)
    {
        num3[i+1] += num3[i] / 10;
        num3[i] %= 10;
    }
    print(num3);
}
```

X

111
78

888
777

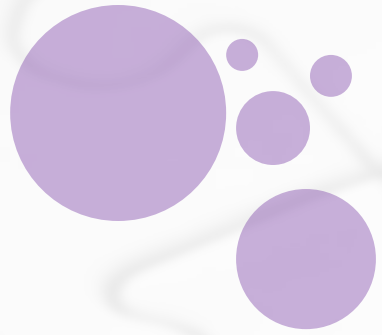


更新

🔲 $10^{10} * 10^{10} = ?$

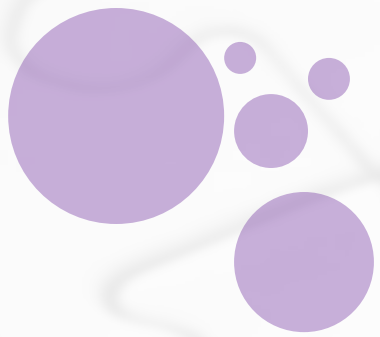
🔲 最高有可能到 10^{20}

🔲 所以 num3 的大小至少要是兩個大數的長度相加



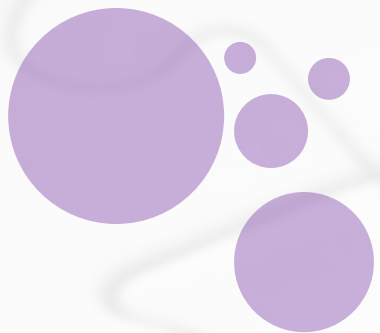
大數運算 – 比大小

- ❖ 比較最高次方的係數
- ❖ 如果最高次方的係數相同，比較第二高的
- ❖ Ex : 12345 ? 67899
- ❖ Ex : 3456 ? 10000



大數運算 – 比大小

```
int compare(int num1[], int num2[])
{
    int i = LEN - 1;
    while(i > 0 && num1[i] == num2[i])
    {
        i--;
    }
    return num1[i] - num2[i];
}
```



其實

🔲在不考慮時間以及效能下

🔲 $9 * 3 = 9 + 9 + 9$

🔲 $9 / 3 = 9 - 3 - 3 - \dots$

🔲Q：所以大數的除法可以怎麼寫？

題外話 – java的大數？

🧠 java提供了一個資料型態可以存放大數類別

```
public class Main {  
    public static void main(String[] argc)  
    {  
        BigInteger a = new BigInteger.ZERO;  
        BigInteger btwo = new BigInteger("2");  
        System.out.printf("%s+1=%s\n", a, a.add(BigInteger.ONE));  
        System.out.printf("%s-1=%s\n", a, a.subtract(BigInteger.ONE));  
        System.out.printf("%s*2=%s\n", a, a.multiply(btwo));  
        System.out.printf("%s/2=%s\n", a, a.divide(btwo));  
    }  
}
```

🧠 好語言不學嗎？



THANK YOU