

my coding environment :

OS:windos

Ide:vs code

Problem in task1:

1. 如何使用雙指標表示二維陣列 及清空其記憶體

Ref: <http://www.programmer-club.com.tw/ShowSameTitleN/c/38340.html>

>2. 用 A[row*N+col] 的方式存取陣列元素感到很不直覺.

這是方法之一, 一次過配置整片記憶體, 用一維的方式來實施二維的概念.

另一個方法是用雙指標來模擬 2 維陣列的操作:

```
int **A ;
A = malloc(N * sizeof *A) ;
for (n = 0 ; n < N ; ++n)
{
    A[n] = malloc(M * sizeof **A) ;
}
```

這樣, A 可以用二維陣列的語法:

```
for (n = 0 ; n < N ; ++n)
{
    for (m = 0 ; m < M ; ++m)
    {
        A[n][m] = ... ;
    }
}
```

釋放記憶體時, 次序要顛倒:

```
for (n = 0 ; n < N ; ++n)
{
    free(A[n]) ;
}
free(A) ;
```

>不知書上所用的整個陣列宣告與元素存取方式有什麼好處?

用 malloc(M * N * sizeof(...)) 比較直接, 配置, 釋放只有一次. 但在使用上不太自然, 因為你必須去寫出計算元素位置的邏輯.

用雙指標在配置, 釋放時手續多一些, 但在使用上會比較自然, 跟固定陣列同樣的語法.

各有各得利弊及使用場地.

2. string.h strtok() 之應用 (為了判斷有無符合 size)

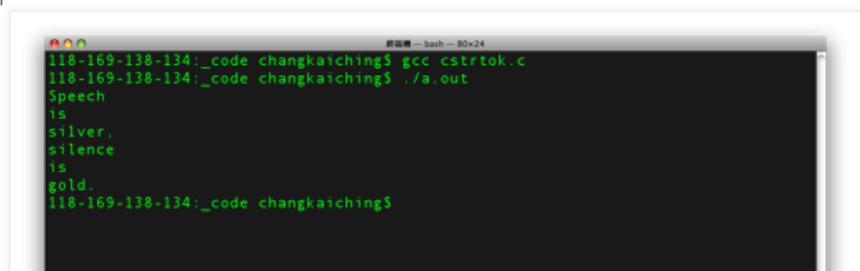
ref: <https://pydoing.blogspot.com/2010/07/c-strtok.html>

string.h 的函數 strtok()，需要兩個字串參數，以第二個參數字串的內容切割第一個參數字串。

以下程式以空格切割字串 s

```
1  #include <stdio.h>
2  #include <string.h>
3
4  int main(void)
5  {
6      char s[] = "Speech is silver, silence is gold.";
7      char t[] = " ";
8      char *test = strtok(s, " ");
9
10     while (test != NULL) {
11         printf("%s\n", test);
12         test = strtok(NULL, " ");
13     }
14
15     return 0;
16 }
17
18 /* 《程式語言教學誌》的範例程式
19    http://pydoing.blogspot.com/
20    檔名: cstrtok.c
21    功能: 示範 string.h 中函數 strtok() 的使用
22    作者: 張凱慶
23    時間: 西元2010年6月 */
```

編譯後執行，結果如下



```
118-169-138-134:~_code changkaiching$ gcc cstrtok.c
118-169-138-134:~_code changkaiching$ ./a.out
Speech
is
silver.
silence
is
gold.
118-169-138-134:~_code changkaiching$
```

3. fgets 的用法 (原因如上)

ref: <http://hsian-studio.blogspot.com/2008/09/fgets.html>

```
char* gear_fgets(char* buf, int num, FILE* fp, bool ignore)
{
    char* find = 0;
    if (!fgets(buf, num, fp))
    {
        return NULL;
    }
    if ((find = strrchr(buf, '\n')))
    {
        *find = '\0';
    }
    else if (ignore)
    {
        char ch;
        while (((ch = fgetc(fp)) != EOF) && (ch != '\n'));
    }
    return buf;
}
```

(不過這裡沒有處理\n 是因為 atoi 只

轉連續整數 所以沒差)

Problem in task2:

1. atof 的用法(為了處理四捨五入)

ref:http://tw.gitbook.net/c_standard_library/c_function_atof.html

以下是atof() 函數的聲明。

```
double atof(const char *str)
```

參數

- str -- 這是表示一個浮點數字的字符串。

返回值

這個函數返回轉換後的浮點數作為一個double值。如果有有效的轉換可以執行，它返回零 (0.0) 。

例子

下麵的例子顯示atof() 函數的用法。

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

int main()
{
    float val;
    char str[20];

    strcpy(str, "98993489");
    val = atof(str);
    printf("String value = %s, Float value = %f\n", str, val);

    strcpy(str, "gitbook.net");
    val = atof(str);
    printf("String value = %s, Float value = %f\n", str, val);

    return(0);
}
```

讓我們編譯和運行上麵的程序，這將產生以下結果：

```
String value = 98993489, Float value = 98993488.000000
String value = gitbook.net, Float value = 0.000000
```