

# 計算機程式語言

---

教課教授：謝東儒

助教：蔡詠聿、吳品頤

# Chapter 8\_project 6

The prototypical Internet newbie is a fellow named B1FF, who has a unique way of writing messages. Here's a typical B1FF communique:

```
H3Y DUD3, C 15 R1LLY C00L!!!!!!!!!!!!
```

Write a “B1FF filter” that reads a message entered by the user and translates it into B1FF-speak:

```
Enter message: Hey dude, C is rilly cool
```

```
In B1ff-speak: H3Y DUD3, C 15 R1LLY C00L!!!!!!!!!!!!
```

Your program should convert the message to upper-case letters, substitute digits for certain letters(A->4, B->8, E->3, I->1, O->0, S->5), and then append 10 or so exclamation marks. Hint: Store the original message in an array of characters, then go back through the array, translating and printing character one by one.

# Solution

```
1
2  #include <stdio.h>
3  #include <ctype.h>
4
5  #define MAX_SIZE 100
6
7  int main(void) {
8
9      char c, message[MAX_SIZE] = {0};
10     int i=0, size=0;
11
12     printf("Enter message: ");
13
14     while((c = getchar()) != '\n' && size < MAX_SIZE)
15     {
16
17         for(i=0; i<size; i++){
18
19
20
21
22
23
24
25
26
27
```

# Solution

```
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38     }  
39  
40     printf("In B1FF-speak: %s!!!!!!!!!!!!\n",message);  
41  
42     return 0;  
43 }
```

# Example

```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter message: bananas
In B1FF-speak: 84N4N45!!!!!!!!!!
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter message: If I Can't Have You
In B1FF-speak: 1F 1 C4N'T H4V3 Y0U!!!!!!!!!!
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$
```

# Chapter 8\_project 7

Write a program that reads a 5 x 5 array of integers and then prints the row sums and the column sums:

Enter row 1 : 8 3 9 0 10

Enter row 2 : 3 5 17 1 1

Enter row 3 : 2 8 6 23 1

Enter row 4 : 15 7 3 2 9

Enter row 5 : 6 14 2 6 0

Row totals : 30 27 40 36 28

Column totals : 34 37 37 32 21

# Solution

```
1 // find sums of row
2 // find sums of columns
3
4 #include <stdio.h>
5
6 #define N 10
7
8 int main(void){
9
10     int a[N][N], i, j, sum ;
11
12     for(i=0; i<N; i++){
13         printf("Enter row %d : ", i+1);
14
15         for(j=0; j<N; j++){
16             scanf("%d", &a[i][j]);
17         }
18     }
19
20     printf("\nRow totals : ");
21     for(i=0; i<N; i++){
22         sum = 0;
23         for(j=0; j<N; j++){
24             sum += a[i][j];
25         }
26         printf(" %d", sum);
27     }
28
29     printf("\nColumn totals : ");
30     for(j=0; j<N; j++){
31         sum = 0;
32         for(i=0; i<N; i++){
33             sum += a[i][j];
34         }
35         printf(" %d", sum);
36     }
37
38
39     return 0;
40 }
```

# Example

```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter row 1: 1 2 3 4 5
Enter row 2: 5 4 3 2 1
Enter row 3: 1 1 1 1 1
Enter row 4: 2 2 2 2 2
Enter row 5: 3 3 3 3 3
Row totals: 15 15 5 10 15
Column totals: 12 12 12 12 12
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$
```



# Chapter 8\_project 9

Write a program that generates a "random walk", across a 10 x 10 array. The array will contain characters (all 1 . 1 initially). The program must randomly "walk" from element to element, always going up, down, left, or right by one element. The elements visited by the program will be labeled with the letters A through Z, in the order visited. Here's an example of the desired output :

[illegible]

# Chapter 8\_project 9

*Hint:* Use the `srand` and `rand` functions (see `deal.c`) to generate random numbers.

After generating a number, look at its remainder when divided by 4. There are four possible values for the remainder—0, 1, 2, and 3—indicating the direction of the next move. Before performing a move, check that (a) it won't go outside the array, and (b) it doesn't take us to an element that already has a letter assigned. If either condition is violated, try moving in another direction. If all four directions are blocked, the program must terminate. Here's an example of premature termination:

A	B	G	H	I	.	.	.	.	.
.	C	F	.	J	K	.	.	.	.
.	D	E	.	M	L	.	.	.	.
.	.	.	.	N	O	.	.	.	.
.	.	W	X	Y	P	Q	.	.	.
.	.	V	U	T	S	R	.	.	.
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.

Y is blocked on all four sides,  
so there's no place to put Z.

# Solution

```
// random walk
#include <stdio.h>
#include <stdlib.h>
include <time.h>

#define N 10
define FILLER '.'

int main(void){

    char a[N][N], letter = 'A';
    int x, y, direction, moves_tried, new_x, new_y;

    for(x=0; x < N ; x++){
        for(y=0; y< N ; y++){
            a[x][y] = FILLER;
        }
    }

    srand((unsigned) time(NULL));

    x = 0;
    y = 0;
    a[x][y] = letter++;
    direction = rand() % 4;
    moves_tried = 0;

    while(moves_tried < [ ] && letter <= [ ] ){
        switch((direction + moves_tried) % 4){
            case 0: [ ] break;
            case 1: [ ] break;
            case 2: [ ] break;
            case 3: [ ] break;
        }
    }
```

# Solution

```
36
37     if( [redacted] && [redacted] &&
38         [redacted] && [redacted] &&
39         a[new_x][new_y] == FILLER){
40
41         x = new_x;
42         y = new_y;
43         a[x][y] = letter++;
44         direction = rand() % 4;
45         moves_tried = 0;
46     }else{
47         [redacted];
48     }
49 }
50
51 for(x=0; x<N; x++){
52     for(y=0; y<N; y++){
53         printf("%c", a[x][y]);
54     }
55     printf("\n");
56 }
57
58
59 return 0;
60 }
```

# Example

```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
```

```
A . . . . .
B C . . . . .
. D . . . . .
. E F . Z Y . . . .
. H G . . X . . . .
. I J . V W . . . .
. . K . U T . . . .
. . L . . S . . . .
. . M N O R . . . .
. . . . P Q . . . .
```

```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
```

```
A . . . . .
B . . . . .
C . . . . .
D E F . . . . .
. . G . . . . .
. . H . . . . O P Q
. . I J K L M N S R
. . . . . Z Y T U
. . . . . X W V
. . . . .
```

```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ █
```

# Chapter 8\_project 14

Write a program that reverses the words in a sentence:

Enter a sentence : you can cage a swallow can't you?

Reversal of sentence: you can't swallow a cage can you?

Hint : Use a loop to read the characters one by one and store them in a one-dimensional char array. Have the loop stop at a period, question mark, or exclamation point (the “terminating character,”), which is saved in a separate char variable. Then use a second loop to search backward through the array for the beginning of the last word. Print the last word, then search backward for the next-to-last word. Repeat until the beginning of the array is reached. Finally, print the terminating character.

# Solution

```
1 // reverse sentence
2
3 #include <stdio.h>
4
5 #define MAX_SENTENCE_LEN 80
6
7 int main(void){
8
9     char ch, sentence[MAX_SENTENCE_LEN] = { ' ' } , terminator = '.';
10    int n, i, start, finish;
11
12    printf("Enter a sentence : ");
13    for(n=1; n < MAX_SENTENCE_LEN ; n++){
14        ch = getchar();
15        if( || || ){
16            terminator = ch;
17            break;
18        }
19        sentence[n] = ch;
20    }
21
22    printf("Reversal of sentence : ");
23    finish = n;
24    for(start = finish - 1; start >= 0 ; start--){
25        if(sentence[start] == ' '){
26            for(i = ; i < ; i++){
27                putchar(sentence[i]);
28            }
29            finish = start;
30        }
31    }
32    printf("%c\n", terminator);
33
34    return 0;
35 }
```

# Example

```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter a sentence: You're going out in this weather?
Reversal of sentence: weather this in out going You're?
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter a sentence: She'll go bananas when you tell her the news.
Reversal of sentence: news the her tell you when bananas go She'll.
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ █
```