

# 程式設計

---

助教：李哲源、傅繹嘉

# Chapter 13\_problem 5

Write a program named sum.c that adds up its command-line arguments, which are assumed to be integers.

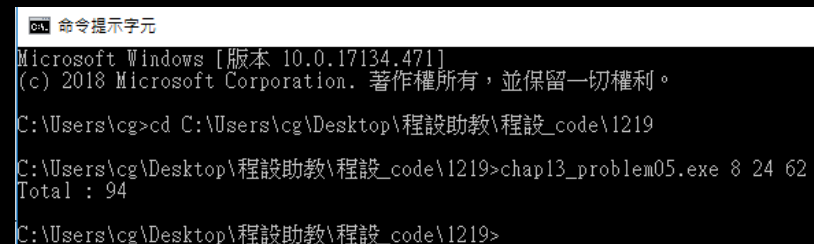
Running the program by typing

`sum 8 24 62`

should produce the following output:

Total: 94

**atoi function** Hint: Use the atoi function to convert each command-line argument from string form to integer form.



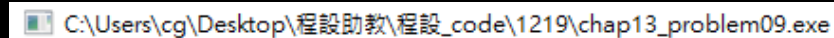
# Solution

```
1 // sum
2
3 #include <stdio.h>
4 #include <stdlib.h>
5
6 int main(int argc, char * argv){
7
8     int i, sum = 0;
9
10    for(i = 1; i < argc; i++){
11        sum += atoi(argv[i]);
12    }
13
14    printf("Total : %d\n", sum);
15
16    return 0;
17 }
18
```

## Chapter 13\_problem 9

Modify Programming Project 10 from Chapter 7 so that it includes the following function: `int compute_vowel_count(const char *sentence);`

The function returns the number of vowels in the string pointed to by the sentence parameter.



C:\Users\cg\Desktop\程設助教\程設\_code\1219\chap13\_problem09.exe

```
Enter a sentence : And that's the way it is.  
Your sentence contains 6 vowels.
```

```
-----  
Process exited after 30 seconds with return value 0  
請按任意鍵繼續 . . .
```

---

# Solution

```
1 // vowels
2
3 #include <ctype.h>
4 #include <stdio.h>
5
6 #define SENTENCE_LEN 80
7
8 int compute_vowel_count(const char *sentence);
9 int read_line(char str[], int n);
10
11 int main(void){
12     char sentence[SENTENCE_LEN + 1];
13
14     printf("Enter a sentence : ");
15
16     read_line(sentence, SENTENCE_LEN);
17
18     printf("Your sentence contains %d vowels.\n", compute_vowel_count(sentence));
19
20     return 0;
21 }
22
23
```

# Solution

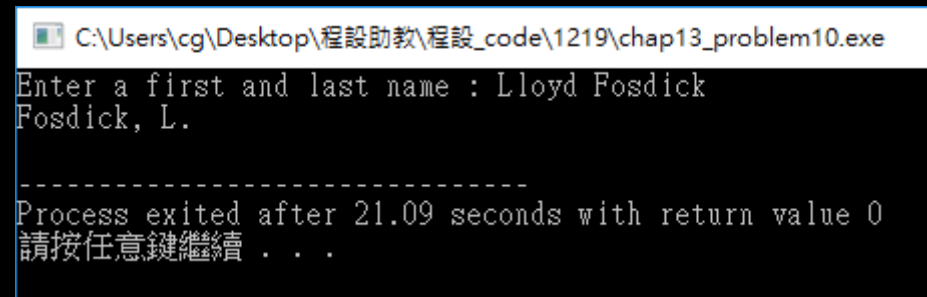
```
24 int compute_vowel_count(const char *sentence){
25
26     int num_vowels = 0;
27
28     while( ) {
29         switch(toupper(*sentence++){
30             case 'A': case 'E': case 'I': case 'O': case 'U':
31                 num_vowels++;
32             }
33         }
34
35     return num_vowels;
36 }
37
38 int read_line(char str[], int n){
39
40     int ch, i=0;
41
42     while((ch = getchar()) != '\n'){
43         if(i < n){
44             str[i++] = ch;
45         }
46     }
47
48     str[i] = '\0';
49
50     return i;
51 }
52
```

# Chapter 13\_problem 10

Modify Programming Project 11 from Chapter 7 so that it includes the following function:

```
void reverse_name(char *name) ;
```

The function expects name to point to a string containing a first name followed by a last name. It modifies the string so that the last name comes first, followed by a comma, a space, the first initial, and a period. The original string may contain extra spaces before the first name, between the first and last names, and after the last name.



```
C:\Users\cg\Desktop\程設助教\程設_code\1219\chap13_problem10.exe
Enter a first and last name : Lloyd Fosdick
Fosdick, L.

-----
Process exited after 21.09 seconds with return value 0
請按任意鍵繼續 . . .
```

# Solution

```
1 // reverse_name
2
3 #include <stdio.h>
4 #include <string.h>
5
6 #define NAME_LEN 30
7
8 void reverse_name(char *name);
9 int read_line(char str[], int n);
10
11
12 int main(void){
13     char name[NAME_LEN + 1];
14
15     printf("Enter a first and last name : ");
16     read_line(name, NAME_LEN);
17
18     reverse_name(name);
19     printf("%s\n", name);
20
21     return 0;
22 }
23
24
```



# Solution

```
24
25 void reverse_name(char *name){
26
27     /* string containing first initial and other formatting */
28     char fi[5];
29     char *p = name;
30
31     /* Find first initial and create first-initial string */
32     while(*p == ' '){
33         p++;
34     }
35     sprintf(fi, ", %c.", *p);
36
37     /* Find Last name */
38     while(*p != ' '){
39         p++;
40     }
41     while(*p == ' '){
42         p++;
43     }
44
45     /* Move Last name to beginning of 'name' string */
46     while(*p != ' ' && *p != '\0'){
47         *name++ = *p++;
48     }
49
50     /* Append first-initial string */
51     strcpy(name, fi);
52 }
53
```

# Solution

```
53
54 = int read_line(char str[], int n){
55     int ch, i = 0;
56
57 =     while((ch = getchar()) != '\n'){
58 =         if(i < n){
59             str[i++] = ch;
60         }
61     }
62
63     str[i] = '\0';
64
65     return i;
66 }
67
```

---