計算機程式語言

教課教授：謝東儒

助教：蔡詠聿、吳品頤

Chapter 13_project 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

```c
//  sum

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

int main(int argc, char * argv[]){

    int i, sum = 0;

    for(i = 1; i <      ; i++){
        sum += atoi(      [i]);
    }

    printf("Total : %d\n", sum);

    return 0;
}

```

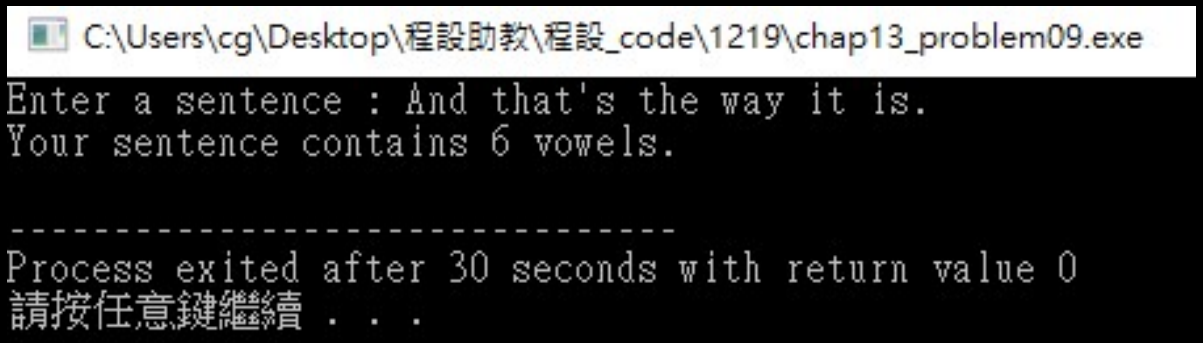
Example

```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out 55 33 44
Total: 132
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out 1 2 3
Total: 6
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out 8 9 10
Total: 27
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$
```

Chapter 13_project 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.



Solution

```c
// vowels

#include <ctype.h>
#include <stdio.h>

#define SENTENCE_LEN 80

int compute_vowel_count(const char *sentence);
int read_line(char str[], int n);

int main(void){

    char sentence[SENTENCE_LEN + 1];

    printf("Enter a sentence : ");

    read_line(sentence, SENTENCE_LEN);

    printf("Your sentence contains %d vowels.\n", compute_vowel_count(          ));

    return 0;
}

```

Solution

```
int compute_vowel_count(const char *sentence){

    int num_vowels = 0;

    while(          ){
        switch(toupper(*sentence++)){
            case 'A': case 'E': case 'I': case 'O': case 'U':
                num_vowels++;
        }
    }

    return num_vowels;
}

int read_line(char str[], int n){

    int ch, i=0;

    while((ch = getchar()) != '  '){
        if(i < n){
            str[i++] = ch;
        }
    }

    str[i] = '  ';

    return i;
}
```

Example

```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter a sentence: aeiouBBbbb
Your sentence contains 5 vowels.
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter a sentence: AEIOUBBbbb
Your sentence contains 5 vowels.
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$
```

Chapter 13_project 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.

Solution

```c
// reverse_name

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

#define NAME_LEN 30

void reverse_name(char *name);
int read_line(char str[], int n);


int main(void){

    char name[NAME_LEN + 1];

    printf("Enter a first and last name : ");
    read_line(name, NAME_LEN);

    reverse_name(name);
    printf("%s\n", name);

    return 0;
}

```

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.", [redacted]);
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, [redacted]);
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
```

Example

```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter a first and last name: Lloyd Posdick
Posdick, L.
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$
```

Chapter 13_project 18

Write a program that accepts a date from the user in the form mm/dd/yyyy and then displays it in the form month dd, yyyy, where month is the name of the month:

```
Enter a date (mm/dd/yyyy): 2/17/2011
You entered the date Februaray 17, 2011
```

Store the month names in an array that contains pointers to strings.

Solution

```c
#include <stdio.h>

int main(void) {

    int m, d, y;

    char [redacted] = {"January", "February", "March", "April", "May", "June",
                       "July", "August", "September", "October", "November",
                       "December"};

    printf("Enter a date (mm/dd/yyyy): ");
    scanf("%d / %d / %d", &m, &d, &y);
    printf("You entered the date %s %.2d, %d\n", [redacted], d, y);
    return 0;
}
```

Example

```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter a date (mm/dd/yyyy): 11/19/1996
You entered the date November 19, 1996
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$
```