

CSE1303 Tutorial Sheet 1 - Week January 7, 2002

Revision

Please attempt all starred questions before your tutorial class.

Exercise 1

Describe the contents of array **a** after the following code is executed.

```
int a[4][4];
int i, j;

for (i = 1; i < 5; i++)
{
    for (j = 1; j < 5; j++)
    {
        a[i-1][j-1] = i*j;
    }
}
```

Exercise 2 *

Write a C function which is given a positive number of seconds, and returns the equivalent hours, minutes and seconds. The values of the minutes and seconds returned must be non negative and less than 60.

Write a main function in C which implements a menu that prompts the user to chose to enter a positive number of seconds or exit the program.

Exercise 3 *

Write fragments of C code showing how you might use the string library functions `strcmp()`, `strcpy()` and `strlen()`.

Exercise 4 *

Write a C function **void reverse(char* s)** which reverses the order of the characters in the character string **s**, and does not call any function from the string library.

Exercise 5

For each of the following, write a single C statement that performs the indicated task.

- (a) Declare and initialise **next** to be a character variable with the value 'A'.
- (b) Declare and initialise **current** to be a character variable with the value 'g'.
- (c) Declare **ptr** to be a pointer to objects of type **char**.
- (d) Assign the address of **current** to the variable **ptr**.
- (e) Change the value of the object pointed to by **ptr** to 'X'.
- (f) Print the address stored in **ptr**.
- (g) Assign the address of **next** to the variable **ptr**.
- (h) Change the value of the object pointed to by **ptr** to 'd'.

What values are stored in **next** and **current**?

Exercise 6 *

Given the following C code.

```
#include <string.h>

#define MAXNAME 80

struct StudentRec
{
    char name[MAXNAME];
    int mark;
};

typedef struct StudentRec Student;
```

Write a C function **void sort(Student class[], int size)** which takes an array of structures of type **Student** and an integer, **size**, which is the size of the array; and sorts the array into alphabetical order according to the names of the students. Write a C function **void printclass(Student class[], char* filename, int size)** to print that array to a file called classlist.txt.

Exercise 7

The following function **strcmp(s, t)**, compares two character strings **s** and **t**, and returns a negative value, zero, or a positive value if **s** is lexicographically less than, equal to, greater than **t**, respectively. Write a pointer version of this function, which doesn't use a variable of type **int**.

```
int
strcmp(char* s, char* t)
{
    int i;

    for (i = 0; s[i] == t[i]; i++)
    {
        if (s[i] == '\\0')
        {
            return 0;
        }
    }

    return s[i] - t[i];
}
```

ADDITIONAL EXERCISES

Deitel & Deitel, Chapter 5

Self-Review Exercises: 5.1 (a to p), 5.2, 5.3, 5.5, 5.7 (a, b, d and e),

Exercises: 5.8, 5.9, 5.10, 5.11, 5.15, 5.16, 5.17, 5.18, 5.19, 5.20, 5.21

Deitel & Deitel, Chapter 7

Self-Review Exercises: 7.1, 7.4, 7.5(a),

Exercises: 7.10

Deitel & Deitel, Chapter 8 (2e)

Self-Review Exercises: 8.1 (a, b, e, f, h, i, o, q, t)

Exercises: 8.9, 8.22, 8.23, 8.24, 8.25,