**Module Assignment 3**

Complete Module Assignment 3 when you finish Learning Package 7.

There are 20 questions in this assignment. All questions carry equal marks.

There is only one correct answer, and you should select the option you

consider the most appropriate.

Write down the missing word or phrase:

1. The use of return transfers control of the program back to where the

function was **c\_\_\_\_d**.

2. A function is called by referring to its name followed by **b\_\_\_\_\_\_s**

enclosing any parameters that are required.

3. A program might use the **l\_\_\_\_\_y** functions getchar() or putchar().

4. The header file stdio.h can be included in the program by the line

**#\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**.

5. A pointer is a symbolic **a\_\_\_\_\_s** in the computer's memory.

6. The address of a variable can be assigned to a **p\_\_\_\_\_r** variable.

7. Each pointer is associated with a particular **d\_\_\_-t\_\_e**.

8. Itis not permissible to use a pointer of one type to **p\_\_\_t** to a variable

of a different data type.

9. In the method of '**call by v\_\_\_e**' information is passed to and from a

function by the values in the argument list.

10. In the method of '**call by r\_\_\_\_\_\_\_e**' the address of a variable is

passed into the function via a pointer within the argument list.

11. An array can be thought of as a variable that is **i\_\_\_\_\_d** to refer to

successive elements.

12. An array is declared by using square brackets which may be empty or

defining the number of **e\_\_\_\_\_ts** in the array.

13. In C the first element of an array is numbered **\_.**

14. Once a pointer is **a\_\_\_\_\_\_d** to an element of an array it can be used

to index the array. This means the pointer can be made to point to any

element of the array by increasing the value in the pointer.

15. Pointer arithmetic works in **u\_\_\_s** that match the size of the data type

they are defined to point to.

16. The variable **s\_\_\_\_** means standard input identify the keyboard.

17. The variable **s\_\_\_\_\_** means standard output identifying the screen.

18. These are stdin and stdout are **f\_\_\_e** pointers.

19. The first parameter in the file versions of the functions fprintf() and

fscanf() is a **p\_\_\_\_\_r** to a file.

20. When using **o\_\_\_\_\_g** a file we need to specify the whether the file is

to be read from, written to, or added to.

**Answers**

1. The use of return transfers control of the program back to where the

function was **called**.

2. A function is called by referring to its name followed by **brackets**

enclosing any parameters that are required.

3. A program might use the **libary** functions getchar() or putchar(). **Linebuffered is also acceptable**.

4. The header file stdio.h can be included in the program by the line

**#include <stdio.h> \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**.

5. A pointer is a symbolic **address** in the computer's memory.

6. The address of a variable can be assigned to a **pointer** variable.

7. Each pointer is associated with a particular **data-type**.

8. It is not permissible to use a pointer of one type to **point** to a variable

of a different data type.

9. In the method of '**call by value**' information is passed to and from a

function by the values in the argument list.

10. In the method of '**call by reference**' the address of a variable is

passed into the function via a pointer within the argument list.

11. An array can be thought of as a variable that is **indexed** to refer to

successive elements.

12. An array is declared by using square brackets which may be empty or

defining the number of **elements** in the array.

13. In C the first element of an array is numbered **0.**

14. Once a pointer is **adressed** to an element of an array it can be used

to index the array. This means the pointer can be made to point to any

element of the array by increasing the value in the pointer.

15. Pointer arithmetic works in **units** that match the size of the data type

they are defined to point to.

16. The variable **stdin** means standard input identify the keyboard.

17. The variable **stdout** means standard output identifying the screen.

18. These are stdin and stdout are **file** pointers.

19. The first parameter in the file versions of the functions fprintf() and

fscanf() is a **pointer** to a file.

20. When using **opening** a file we need to specify the whether the file is

to be read from, written to, or added to.