## Question 1

a)

      i.      List and describe the Windows API calls found in a simple message loop.

      ii.     Under which condition does the message loop break.

(6 marks)

b)

      i.      List and write short notes on the three conditions under which a *process* terminates.

      ii.     List and write short notes on the three conditions under which a *thread* terminates.

(4 marks)

c)  In pseudo-code (but calling the API functions correctly) write down the implementation of a stack data structure using *heap* API calls. Your implementation should include the *Push* and *Pop* functions to add and remove *integer* values on the stack.

(10 marks)

d)  Explain how thread priorities work. In you explanation you should mention *base priorities, relative priorities* and *thread priority boosting in Windows 9x and NT families*.

(5 marks)

**[Total: 25 marks]**

## Question 2

a) Write down the prototype for the entry-point function of a DLL explaining all its arguments.

(4 marks)

b) You are required to add a button control to an existing window whose handle is *hWnd*.

    i.       Write down the API call to create the button and place it on the window.

    ii.      How are messages received by the button control handled in the callback of the main window?

(6 marks)

c)

    i.       List and write down short notes on 4 *mapping modes* in windows.

    ii.      Why are mapping modes important?

(5 marks)

d)

    i.       Why is *thread synchronization* an issue?

    ii.      In pseudo-code write create a scenario that would illustrate such an issue.

    iii.    What technique can be used to solve the problem illustrated in your code above? Use the technique you mention to fix your pseudo-code.

(6 marks)

e)

    i.       What is the purpose of the *lParam* and *wParam* arguments in a Windows callback function? Give examples.

    ii.      When you are creating a window, where do you specify which callback function to use for it?

(4 marks)

**[Total: 25 marks]**

## Question 3

a)

    i.      Write down the *resource script* to create a menu with the structure below (Note the underlined hotkeys):

- File
    - New...
    - Open...
    - Save
    - ------------------
    - Exit
- Edit
    - Cut
    - Copy
    - Paste
- Help
    - About...

    ii.    How is the menu attached to a window?

    iii.   How is a click on each menu item detected by the window?

(8 marks)

b)

    i.      What is the difference between a *COM client* and a *COM server*?

    ii.    List and briefly describe the type types of COM servers.

(4 marks)

c) Write short notes on the following:
    i.      Accelerators.
    ii.    Accelerator tables.
    iii.   Modifying the standard message loop to process accelerators.
    iv.   Handling accelerators in the window callback function.

(10 marks)

d) Briefly describe the purpose of device contacts in Windows.

(3 marks)

**[Total: 25 marks]**