Lab Assignment - 10

Marks:30

1. Define a union named `Data` that can store an integer, a float, and a character. Write a C program to demonstrate how to assign and access values to/from this union.

[3 marks]

2. Given a structure and a union, both containing two integers and a character, write a program to illustrate the difference in memory allocation between the structure and the union. Use the `sizeof` operator to display the size of the structure and the union. Explain how the memory allocation for unions differs from structures.

[3 marks]

3. Create a structure named `Employee` that contains an employee's ID (as an integer), name (as a character array), and a union named `EmpDetails`. The union `EmpDetails` should contain an employee's hourly wage (as a float) and a fixed salary (as a double). Write a program that allows assigning and accessing these elements. Explain when you might prefer using unions within structures.

[4 marks]

4. Define a union that can store an array of integers, an array of floats, and an array of characters. Write a program that demonstrates how to assign values to each array within the union and then access and print these values. Discuss the limitations you encounter when working with arrays within unions.

[4 marks]

5. Expand the `Employee` structure from Question 3 to include an enumeration named `PayType` with two values: `HOURLY` and

`SALARY`. Modify the union `EmpDetails` to also include this enumeration. Write a program that assigns appropriate values based on the pay type and then accesses and prints these values. Discuss how enumerations can help manage unions more effectively.

[6 marks]

6. Consider a scenario where you need to design a data packet for a networking application that can contain either a control message or data payload. The control message consists of an opcode (as an integer) and a status code (as an integer), whereas the data payload is a string of characters.

      a. Define a union named `Packet` that can store both the control message and data payload.

      b. Define a structure named `DataPacket` that includes a flag indicating the type of packet (control or data) and the union `Packet`.

      c. Write a C program that creates an array of `DataPacket`, assigns values to demonstrate both control and data packets, and then prints these packets based on the type.

In your program, include functions for assigning values to the packets and for printing the packet contents. Discuss how using unions in this scenario provides flexibility and efficient memory usage for varying data types within the packets.

[10 marks]