Date: / /

Lab Practical #14:

Implementation of parity bit check Using C/Java language with example.

Practical Assignment #14:

1. **C/Java Program: Implementation of parity bit check Using C/Java language.**

```
#include <stdio.h>
int calculateParity(int data[], int size, int isEvenParity)
{
    int count = 0;
    for (int i = 0; i < size; i++){
        if (data[i] == 1){
            count++;
        }
    }
    if (isEvenParity){
        return (count % 2 == 0) ? 0 : 1;
    }
    else{
        return (count % 2 == 0) ? 1 : 0;
    }
}
int verifyParity(int data[], int size, int receivedParityBit, int isEvenParity){
    int oneCount = 0;
    for (int i = 0; i < size; i++){
        if (data[i] == 1){
            oneCount++;
        }
    }
    if (receivedParityBit == 1){
        oneCount++;
    }
    if (isEvenParity){
```

Date: / /

```
        return (oneCount % 2 == 0);
    }
    else{
        return (oneCount % 2 != 0);
    }
}
int main()
{
    int data[] = {1, 0, 1, 1, 0, 1};
    int size = sizeof(data) / sizeof(data[0]);
    int isEvenParity;
    printf("Choose the parity type:\n");
    printf(" Enter 1 for Even Parity\n");
    printf(" Enter 0 for Odd Parity\n");
    printf("Your choice: ");
    scanf("%d", &isEvenParity);
    if (isEvenParity != 0 && isEvenParity != 1){
        printf("Invalid choice. Please run again and enter 0 or 1.\n");
        return 1;
    }
    printf("\nOriginal Data: ");
    for (int i = 0; i < size; i++){
        printf("%d", data[i]);
    }
    printf("\n");
    printf("Parity Type Selected: %s\n", isEvenParity ? "Even" : "Odd");
    int parityBit = calculateParity(data, size, isEvenParity);
    printf("Sender: Calculated Parity Bit is %d\n", parityBit);
    printf("Sender: Full message to send is ");
    for (int i = 0; i < size; i++){
        printf("%d", data[i]);
    }
```

Date: / /

```
    printf("%d\n", parityBit);
    printf("\n--- Simulating Data Reception ---\n");
    int receivedData[size];
    printf("Enter the %d bits of the data you 'received':\n", size);
    for (int i = 0; i < size; i++){
        printf("Enter bit %d: ", i + 1);
        scanf("%d", &receivedData[i]);
    }
    int receivedParityBit = parityBit;
    printf("\nReceiver: Checking received message ");
    for (int i = 0; i < size; i++){
        printf("%d", receivedData[i]);
    }
    printf("%d\n", receivedParityBit);
    if (verifyParity(receivedData, size, receivedParityBit, isEvenParity)){
        printf("Result: Parity check PASSED.(Data is considered correct)\n");
    }
    else{
        printf("Result: Parity check FAILED. (Error detected)\n");
    }
    return 0;
}
```

Output:

```
Choose the parity type:
 Enter 1 for Even Parity
 Enter 0 for Odd Parity
Your choice: 1

Original Data: 101101
Parity Type Selected: Even
Sender: Calculated Parity Bit is 0
```

Date: / /

Sender: Full message to send is 1011010

--- Simulating Data Reception ---
Enter the 6 bits of the data you 'received':
Enter bit 1: 1
Enter bit 2: 0
Enter bit 3: 1
Enter bit 4: 1
Enter bit 5: 0
Enter bit 6: 1

Receiver: Checking received message 1011010
Result: Parity check PASSED.(Data is considered correct)