North West Power Generation Company Limited (NWPGCL)
Post: AM (ICT), Date: 12/01/2024, Centre: BUET Non: 60, Dept:40, Time: 1 hr

1. Salary Range and Tax Calculation are given.

Salary Range	Tax
0-250000	**0**
250001-5000000	**10%**
500001-100000	**20%**
>10,00000	**30%**

a. Write a program using any language to the calculate the total tax of employee salary

```
#include <stdio.h>

  int main() {
   // Constants for tax ranges
   const int range1 = 250000;
   const int range2 = 500000;
   const int range3 = 1000000;
   // Constants for tax rates
   const float rate1 = 0.0;
   const float rate2 = 0.10;
   const float rate3 = 0.20;
   const float rate4 = 0.30;
   // Number of employees
   int numEmployees;
   printf("Enter the number of employees: ");
   scanf("%d", &numEmployees);
   // Array to store employee salaries
   float salaries[numEmployees];
   // Input employee salaries
   printf("Enter the salaries of employees:\n");
   for (int i = 0; i < numEmployees; i++) {
      printf("Employee %d: ", i + 1);
```

```
        scanf("%f", &salaries[i]);
    }
    // Calculate tax and total tax
    float totalTax = 0.0;
    for (int i = 0; i < numEmployees; i++) {
        float tax = 0.0;
        if (salaries[i] <= range1) {
            tax = salaries[i] * rate1;
        } else if (salaries[i] <= range2) {
            tax = range1 * rate1 + (salaries[i] - range1) * rate2;
        } else if (salaries[i] <= range3) {
            tax = range1 * rate1 + (range2 - range1) * rate2 + (salaries[i] - range2) *
rate3;
        } else {
            tax = range1 * rate1 + (range2 - range1) * rate2 + (range3 - range2) * rate3
+ (salaries[i] - range3) * rate4;
        }

        totalTax += tax;
        printf("Employee %d: Salary = %.2f, Tax = %.2f\n", i + 1, salaries[i], tax);
    }

    printf("Total Tax for all employees: %.2f\n", totalTax);

    return 0;
}
```

Output:

```
Enter the number of employees: 5
Enter the salaries of employees:
Employee 1: 25000
Employee 2: 250000
Employee 3: 300000
Employee 4: 6000000
Employee 5: 700000
Employee 1: Salary = 25000.00, Tax = 0.00
Employee 2: Salary = 250000.00, Tax = 0.00
Employee 3: Salary = 300000.00, Tax = 5000.00
Employee 4: Salary = 6000000.00, Tax = 1625000.00
```

Employee 5: Salary = 700000.00, Tax = 65000.00
Total Tax for all employees: 1695000.00

b. From the three employee salary find the highest tax paying employee

```
#include <stdio.h>

int main() {
    // Constants for tax ranges
    const int range1 = 250000;
    const int range2 = 500000;
    const int range3 = 1000000;

    // Constants for tax rates
    const float rate1 = 0.0;
    const float rate2 = 0.10;
    const float rate3 = 0.20;
    const float rate4 = 0.30;

    // Number of employees
    const int numEmployees = 3;

    // Array to store employee salaries
    float salaries[numEmployees];

    // Input employee salaries
    printf("Enter the salaries of three employees:\n");
    for (int i = 0; i < numEmployees; i++) {
        printf("Employee %d: ", i + 1);
        scanf("%f", &salaries[i]);
    }

    // Calculate tax and find the highest tax paying
employee
    float maxTax = 0.0;
    int highestTaxEmployee = 0;
```

```
    for (int i = 0; i < numEmployees; i++) {
        float tax = 0.0;

        if (salaries[i] <= range1) {
            tax = salaries[i] * rate1;
        } else if (salaries[i] <= range2) {
            tax = range1 * rate1 + (salaries[i] -
range1) * rate2;
        } else if (salaries[i] <= range3) {
            tax = range1 * rate1 + (range2 - range1) *
rate2 + (salaries[i] - range2) * rate3;
        } else {
            tax = range1 * rate1 + (range2 - range1) *
rate2 + (range3 - range2) * rate3 + (salaries[i] -
range3) * rate4;
        }

        if (tax > maxTax) {
            maxTax = tax;
            highestTaxEmployee = i;
        }

        printf("Employee %d: Salary = %.2f, Tax =
%.2f\n", i + 1, salaries[i], tax);
    }

    printf("Employee %d has the highest tax payment:
%.2f\n", highestTaxEmployee + 1, maxTax);

    return 0;
}
```

Output:

```
Enter the salaries of three employees:
Employee 1: 300000
Employee 2: 450000
Employee 3: 600000
Employee 1: Salary = 300000.00, Tax = 5000.00
Employee 2: Salary = 450000.00, Tax = 20000.00
Employee 3: Salary = 600000.00, Tax = 45000.00
Employee 3 has the highest tax payment: 45000.00
```

2.

Product (pname,price,category,maker)

Purchase (buyer,seller,stock)

Company(cname,country)

Person(per-name, address, phone)

Here in Product table make is similar to cname, here buyer and seller in Purchase table is similar to per-name in Person table,

Draw the ER diagram of from the above information

(a) Select person name who bought American category product

```
SELECT per-name
FROM Person
WHERE EXISTS (
SELECT 1
FROM Purchase
WHERE Purchase.buyer = Person.per-name
AND EXISTS (
SELECT 1
 FROM Product
 WHERE Product.pname = Purchase.seller
 AND Product.category = "American"
 )
);
```

(b) Find Sum of all product and Sum of All sales product according to country of origin.

```
SELECT Company.country,
SUM(Product.price) AS total_product_price,
SUM(Purchase.price) AS total_sales_price
FROM Product
INNER JOIN Purchase ON Product.pname = Purchase.seller
INNER JOIN Company ON Product.maker = Company.cname
GROUP BY Company.country;
```

3.Given IP address 192.168.1.50, Subnet Mask: 255.255.255.240. find the valid IP range. Also find Network address and Broadcast address.

Convert IP and Subnet Mask to Binary:

Convert the IP address and subnet mask to binary form.

IP Address (192.168.1.50) in binary: 11000000.10101000.00000001.00110010

Subnet Mask (255.255.255.240) in binary: 11111111.11111111.11111111.11110000

Identify Network Portion:

Identify the network portion by performing a bitwise AND operation between the IP address and the subnet mask.

192.168.1.50 AND 255.255.255.240 = 192.168.1.48

So, the network portion is 192.168.1.48.

Find Broadcast Address:

To find the broadcast address, perform a bitwise OR operation between the network portion and the inverted subnet mask.

192.168.1.48 OR 0.0.0.15 = 192.168.1.63

So, the broadcast address is 192.168.1.63.

Find Valid IP Range:

The valid IP range is from the network address + 1 to the broadcast address - 1.

Valid IP range: 192.168.1.49 to 192.168.1.62

Therefore, the requested information is as follows:

Network Address: 192.168.1.48

Broadcast Address: 192.168.1.63

Valid IP Range: 192.168

If You are CEO of a software company. You need to develop an ERP software from following three options (i) Buy (ii) Build (iii) Open Source Modification
a) Buy: Buy a software with cost 50 lac.
b) Building: Developed by developer cost 40 lac for easy process. 50 lac for hard process. Possibility is 30% to develop in easy process.
c) Open Source and Modification : Buy and small modifications cost 30 lac, for large modifications cost 50 lac. Possibility is 80% for large.

What way you choose and why? Explain with calculation.

Here's my analysis of the options, incorporating calculations and considerations:

Option A: Buy

- Cost: 50 lac (fixed)
- Pros:
  - Quickest implementation
  - Established reliability
  - Vendor support
- Cons:
  - Limited customization
  - Potential vendor lock-in

Option B: Build

- Cost:
  - Expected cost: (0.3 * 40 lac) + (0.7 * 50 lac) = 47 lac
  - Worst-case cost: 50 lac
- Pros:
  - Tailored to specific needs

- Full control over development

- Cons:
  - Longer development time
  - Uncertain cost and timeline
  - Potential for scope creep

Option C: Open Source and Modification

- Cost:
  - Expected cost: (0.2 * 30 lac) + (0.8 * 50 lac) = 46 lac
  - Worst-case cost: 50 lac
- Pros:
  - Balance of customization and cost
  - Access to community resources
  - Potential for continuous improvement
- Cons:
  - Dependency on external project
  - Integration challenges

Decision and Reasoning:

I would choose Option C: Open Source and Modification.

Here's why:

- Cost: It offers a potentially lower expected cost (46 lac) compared to buying (50 lac) or building (47 lac worst-case).
- Customization: It provides a degree of customization, though not as extensive as building from scratch.

- Time: It can be implemented more quickly than building, though potentially slower than buying.
- Community and Support: It benefits from a community of developers, resources, and potential support.
- Risk Mitigation: It balances the risks of cost overruns in building and vendor lock-in in buying.

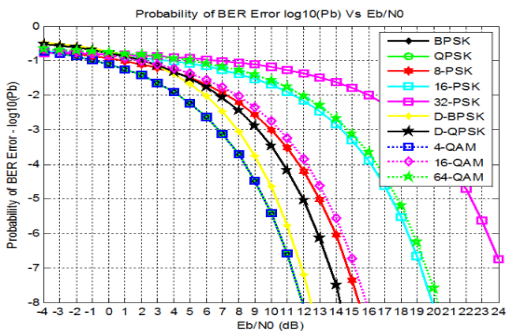
Additional Considerations:

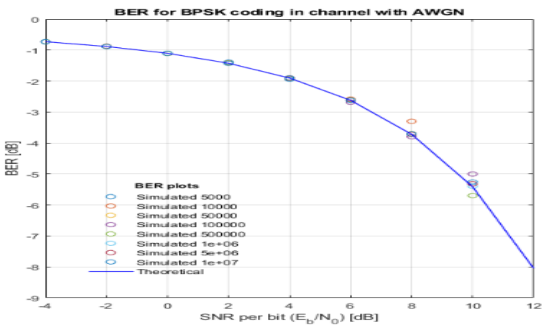
- Specific Requirements: If the software needs highly unique or sensitive features, building might be necessary.
- Internal Expertise: If the company has strong development capabilities, building could be more feasible.
- Vendor Reputation: If a reputable vendor with excellent support is available for Option A, it could be compelling.
- Open Source Community: The health and activity of the open source project in Option C should be carefully evaluated.

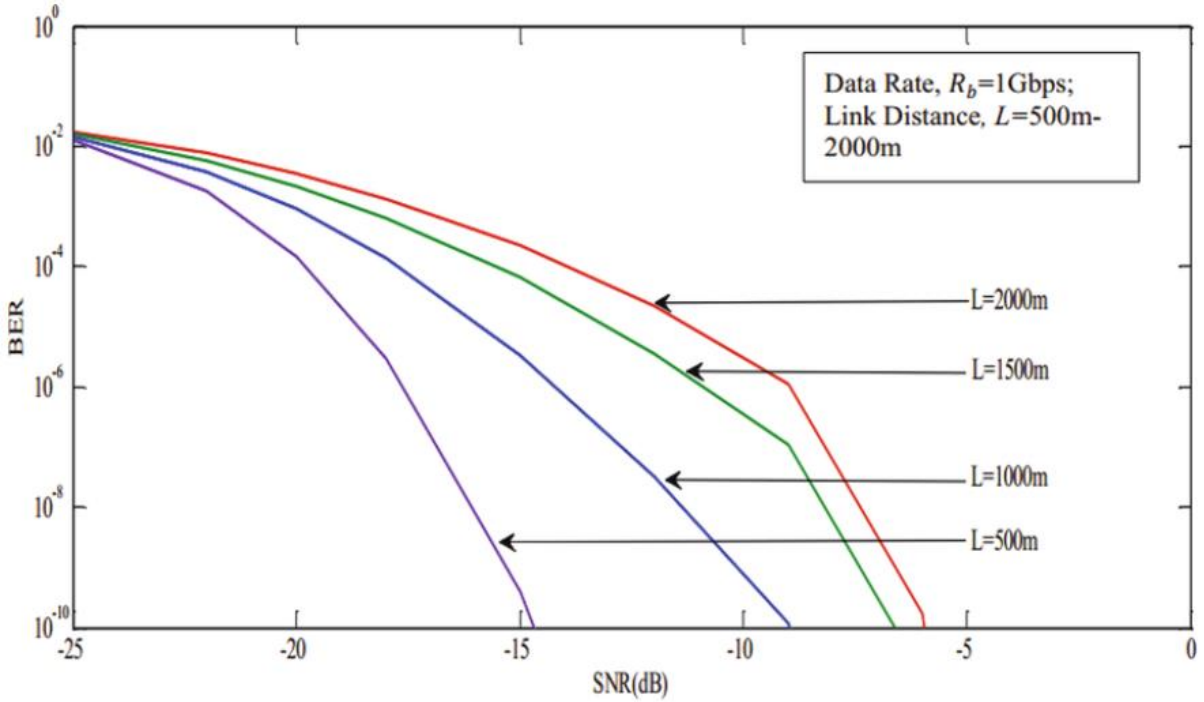
Conclusion:

While the decision ultimately depends on specific company priorities and project needs, Option C presents a balanced approach in terms of cost, customization, time, risk, and community support. It's worth thoroughly evaluating all options, considering both quantitative and qualitative factors, to make the most informed decision.

5.Draw Bit Error Rate vs Signal to Noise Ratio curve of QPSK and BPSK .






Data Rate, R_b=1Gbps;
Link Distance, L=500m-2000m
L=2000m
L=1500m
L=1000m
L=500m
BER
SNR(dB)
10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
-25
-20
-15
-10
-5
0