# MINI PROJECT QUESTIONS

Design a compiler(Lexical  and Parser phase) for the
following hypothetical languages.

Q1:

```
int main()
begin
    int n, i, sum = 0;
        for(i=1; i <= n; ++i)
          begin
             expr= expr+expr;
        end
End
```

Q2:

```
int main()
begin
    int n1, n2, n3;
     if( expr relop expr )
         begin
       printf( n1);
    end
    if ( expr relop expr )
    begin
        printf( n2);
    end
    if( expr relop expr )
    begin
        printf( n3);
    end
end
```

Q3:

```
int main()
begin
    int n, re = 0, rem;
    while(expr)
    begin
       expr=expr+expr;
    end
end
```

Q4:

```
int main()
begin
    int n1, n2, i, gcd;
    if(expr relop expr)
       gcd = i;
    for(i=1; expr relop expr; ++i)
    begin
        gcd=1;
    end
end
```

Q5:

```
BEGIN
    PRINT "HELLO"
    INTEGER A, B, C
    REAL   D, E
    STRING X, Y
    A := 2
    B := 4
    C := 6
    D := -3.56E-8
    E := 4.567
```

```
    X := "text1"
    Y := "hello there"

    FOR I:= 1 TO 5
            PRINT "Strings are [X] and [Y]"
      END
```

Q6:

```
X: integer ;

Procedure  foo( b : integer )
b := 13;
If x = 12 and b = 13 then
printf( "by copy-in copy-out" );
elseif x = 13 and b = 13 then
printf( "by address" );
else
printf( "A mystery" );
end if;
end foo
```

Q7:

```
int main()
begin
   int count=1;
   while(n>1)
      count=count+1;
      n=n/2;
   end while
 return count
end
```

Q8:

```
int main()
begin
    int L[10];
    int maxval=L[0];
    for i=1 to n-1 do
       if L[i]>maxval
           maxval=L[i];
       endif
    endfor
    return(maxval)
End
```

Q9:

```
int main()
begin
int n;
do
        expr=expr+expr;
        n=exp;
 while(exp)
 return(n)
end
```

Q10:

```
int main()
    char operator;
    int firstNumber,secondNumber;

    switch(operator)
    begin
      case '+':
            printf(firstNumber+secondNumber);
            break;

        case '-':
```

```
            printf(firstNumber-secondNumber);
            break;
        end
    return 0;
```