NAME- RUTVIK MARAKANA

ASSIGNMENT – 4

1. *Problem 1 solution*:-

```
CLASS SandClock
BEGIN

    METHOD Main
    BEGIN

        FOR (i←5; i>=1; i--)

            FOR (j←i; j<9;j++)

                PRINT  “  ”
            ENDFOR
            FOR (k←1;k<=2*i-1;k++)

                PRINT *
            ENDFOR
        ENDFOR

        FOR (i←2; i<=5; i++)

            FOR (j←i; j<9 ; j++)

                PRINT “ ”
            ENDFOR
            FOR (k←1; k<=2*i-1; k++)

                PRINT *
            ENDFOR
        ENDFOR
```

END Main

END SandClock

2. *Problem 2 solution:-*

```
CLASS DailyCaloriesAllowed
BEGIN

    METHOD Main
    BEGIN

        READ user_input values for height(in inches), weight(in pounds),
        age, gender and the intensity of exercise from the user.

      height←user_input for the height
     weight←user_input for the weight
         age←user_input for the age
    gender←user_input for the gender
    exercise←user_input for the intensity of exercise

     IF(gender = Female)THEN

         BMR←655 + (4.35 * weight) + (4.7 * height) - (4.7 * age )
    ELSE

         BMR←66 + (6.23 * weight) + (12.7 * height) - (6.8 * age)
    ENDIF

    CASE exercise OF
      1: DCA←BMR*1.2
      2: DCA←BMR*1.375
      3: DCA←BMR*1.55
      4: DCA←BMR*1.725
      5: DCA←BMR*1.9
    ENDCASE
```

```
            PRINT  gender , height inches , weight lbs , Age  age , BMR= BMR
                  , Exercise exercise , DCA:  DCA

    END Main

END DailyCaloriesIntake
```

3. *Problem 3 solution:-*

```
CLASS Pattern
BEGIN

    METHOD Main
    BEGIN

        FOR (i←6; i>=1 ; i--)

            FOR ( j←6; j>i ; j--)

               PRINT " "
            ENDFOR

            FOR (k←1; k<=i; k++)

               PRINT *
            ENDFOR
        ENDFOR
    END Main
END Pattern
```

4. *Problem 4 solution*

```
CLASS LargestOccurenceDigit
BEGIN

        METHOD Main
        BEGIN

            READ user_input for the number of integers. Ask the user to
            enter 0 to exit.
            n←user_input for the number of intgers
            max←0
            occurrence←0

            FOR(i←0;i<n;i++)

                READ user_input for a positive integer. Ask the user to enter
                0 to exit.
                num←user_input for the integer

               IF (num!=0 && num>0)

                   IF(num>max)THEN

                       max←num
                       occurrence=1

                   ELSE
                      occurrence←occurrence+1
                   ENDIF
               ENDIF
         ENDFOR

        PRINT  Largest number:   max
        PRINT  Occurrence   occurrence   times
    END Main
END LargestOccurenceDigit
```