Author: Adrian Szklarski, 30.06.2022.

Technological level: Advanced +

Programming level: Mid

Technologies used: Python 3.10 | PyCharm | Linux | OpenOffice

Description of the problem: In an old castle there is a mysterious room of 60 m^2 which is covered with dust, cobwebs and is dark. Once, at midnight, the ghost of a man who hanged himself for love in the 11th century entered the room. He was unhappy. There was one window open and closed in the room. At the moment of his entrance it opened for a moment so that 25% of the air escaped and 25% of fresh air came inside. A ghost is made of matter which generates heat to such an extent that it replenishes the heat loss due to heat penetration through the walls, but also heats up the exchanged air. The wretch is also made up of water just like man, and this water evaporates. The question is: how much must the poor ghost evaporate and how much heat must it supply to the castle room so that by 1 a.m. when it disappears the air returns to its initial state i.e. temperature t_p = 20 °C and relative humidity φ_p = 70%. Outside air conditions: t_0 = 0 oC and φ_0 = 80%. Brrrr... sounds awful....

1. We use the graph: $i_{(1+x)} = f(x) \rightarrow \mathbf{x_p}$ [g/kg gs], $\mathbf{i_p}$ [kJ/ kg gs], $\mathbf{x_0}$ [g/kg gs], $\mathbf{i_0}$ [kJ/ kg gs];

2. $x_M = \frac{L_p x_p + L_0 x_0}{L_p + L_0}$ [g/ kg gs] , $i_M = \frac{L_p i_p + L_0 i_0}{L_p + L_0}$ [kJ/ kg gs]

3. For our point M from the *i-x* plot we read t_M and φ_M , interpolating we get φ;

4. Room air mass: $m_r = V\, ro$

5. Amount of evaporated from ghost water: $m_{h2o} = m_r (x_w - x_M)$

6. Amount of additional heat input $Q_g = m_r (i_w - i_M)$

7. The entire process is characterized by simple directionality: $\frac{Di}{Dx}$ [kJ/kg]

```
'''   Szklarski Adrian 06.2022


How much must the poor ghost evaporate and how much heat must it supply
to the castle room so that by 1 a.m. when it disappears the air returns
to its initial state i.e. temperature tp = 20 °C and relative humidity
φp = 70%. Outside air conditions: t0 = 0 oC and φ0 = 80%.


    Returns:      - Amount of evaporated from ghost water;
                  - Amount of additional heat input;
                  - The entire process is characterized by simple directionality.


    Parameters:   - Data of task: V, %, tp, fi_w, t0, fi_z;
                  - Chart data: xw, iw, x2, iz;
                  - Point M: tM, fiM, fi.
'''


# data of task: V, %, tp, fi_w, t0, fi_z
data_of_task = [60, 25, 20, 70, 0, 80]
# chart data: xw, iw, x2, iz
data = [10.4, 47, 3.1, 8]
# point M: tM, fiM, fi
pointM = [15, 0.87, 1.21]



def ghost(Data, Data_of_task):
```

```
    sum_air = (Data_of_task[1] + (100 - Data_of_task[1]))
    iM = (Data_of_task[1] * Data[3] + ((100 - Data_of_task[1]) * Data[1])) / sum_air
    xM = (Data_of_task[1] * Data[2] + ((100 - Data_of_task[1]) * Data[0])) / sum_air
    m_room = Data_of_task[0] * pointM[2]
    h2o = round(m_room * (Data[0] - xM), 2)
    Qg = round(m_room * (Data[1] - iM), 2)
    charakteristic = round((1000 * Qg) / h2o, 2)
    return h2o, Qg, charakteristic


hear = ghost(data, data_of_task)
print('       Amount of evaporated from ghost water: ', hear[0], '[g]', '\n\
      Amount of additional heat input: ', hear[1], '[kJ/h]', '\n\
      The entire process is characterized by simple directionality: ', hear[2], '[kJ/kg]')
```

```
Amount of evaporated from ghost water:  132.5 [g]
Amount of additional heat input:  707.85 [kJ/h]
The entire process is characterized by simple directionality:  5342.26 [kJ/kg]
```