

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное автономное образовательное
учреждение высшего образования
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**



Инженерная школа природных ресурсов


Направление подготовки 18.04.01 «Химическая технология»

Образовательная программа «Химическая технология подготовки нефти и газа»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1

По дисциплине
РУТНОН ДЛЯ ЗАДАЧ ХИМИЧЕСКОЙ ТЕХНОЛОГИИ

Студент

Группа	ФИО	Подпись	Дата
2ДМ22	Лукьянов Д.М.		11.11.2023

Руководитель

Должность	ФИО	Ученая степень, звание	Подпись	Дата
доцент ОХИ ИШПР	Чузлов В.А.	к.т.н.		13.11.2023

ЗАДАНИЕ 1

Используя исходные данные из примера, рассчитайте, реализовав соответствующие функции:

1. Состав потока в мольных долях:

$$\chi_i = \frac{\frac{\omega_i}{M_i}}{\sum_{i=1}^n \frac{\omega_i}{M_i}}$$

где χ_i – мольная доля i -го компонента;
 ω_i – массовая доля i -го компонента;
 M_i – молярная масса i -го компонента;
 n – число компонента в системе;
 i – индекс компонента в системе.

2. Плотность потока:

$$\rho = \frac{1}{\sum_{i=1}^n \frac{\omega_i}{\rho_i}}$$

где ρ – плотность потока;
 ω_i – массовая доля i -го компонента;
 ρ_i – плотность i -го компонента;
 n – число компонента в системе;
 i – индекс компонента в системе.

3. Средняя молекулярная масса потока:

$$m = \frac{1}{\sum_{i=1}^n \frac{\omega_i}{M_i}}$$

где m – средняя молекулярная масса потока;
 ω_i – массовая доля i -го компонента;
 M_i – молярная масса i -го компонента;
 n – число компонента в системе;
 i – индекс компонента в системе.

РЕШЕНИЕ 1

Программная реализация:

```
def calc_mole_fracs(
    mass_fractions: list[float],
    densities: list[float],
    mms: list[float]
) -> list[float]:

    mass_frac_by_mms = [mf / mm for mf, mm in zip(mass_fractions, mms)]
    s = sum(mass_frac_by_mms)

    return [x / s for x in mass_frac_by_mms]

def calc_dens(
    mass_fractions: list[float],
    densities: list[float]
) -> list[float]:

    return 1/sum([mf / dens for mf, dens in zip(mass_fractions, densities)])

def calc_mm_mix(
    mass_fractions: list[float],
    mms: list[float]
) -> list[float]:

    return 1/sum([mf / mm for mf, mm in zip(mass_fractions, mms)])
```

```
mf = [.1, .1, .1, .4, .2, .05, .03, .02]
rho = [.416, .546, .585, .5510, .6, .616, .6262, .6594]
mms = [16, 30, 44, 58, 58, 72, 72, 86]

mole_fracs = calc_mole_fracs(mf, rho, mms)

print('Мольные доли:')
for mole_frac in mole_fracs:
    print(f'{mole_frac:.4f}')
print(f'Сумма = {sum(mole_fracs):.4f}')
```

```

dens_mix = calc_dens(mf, rho)
print(f'Плотность {dens_mix:.4f} кг/м3')

mm_mix = calc_mm_mix(mf, mms)
print(f'Молекулярная масса = {mm_mix:.2f} г/моль')

```

Ответ:

Мольные доли:

0.2655

0.1416

0.0965

0.2929

0.1465

0.0295

0.0177

0.0099

Сумма = 1.0000

Плотность 0.5515 кг/м3

Молекулярная масса = 42.47 г/моль

ЗАДАНИЕ 2

Пусть на смешение поступают материальные потоки следующего состава (массовые доли):

Поток	C_1	C_2	C_3	iC_4	nC_4	iC_5	nC_5	C_6
1	0.1	0.1	0.1	0.4	0.2	0.05	0.03	0.02
2	0.1	0.2	0.1	0.3	0.1	0.15	0.03	0.02
3	0.1	0.1	0.15	0.35	0.1	0.05	0.08	0.07

Расходы потоков 200, 250 и 120 кг/ч, соответственно. Необходимо рассчитать состав итогового потока в массовых долях, реализовав соответствующую функцию.

Состав смешиваемого потока можно найти следующим образом:

$$\omega_i = \frac{\sum_{j=1}^n G_j \cdot \omega_{i,j}}{\sum_{j=1}^n G_j}$$

где ω_i – массовая доля i -го компонента в смешиваемом потоке;

$\omega_{i,j}$ – массовая доля i -го компонента в j -ом потоке;

G_j – массовый расход j -го потока;

j – индекс потока;

i – индекс компонента в системе;

n – число потоков, подаваемых на смешение;

РЕШЕНИЕ 2

Программная реализация:

```
def calc_mix_comp(
    comp_flows: list[list[float]],
    flows: list[float]
) -> list[float]:

    comp_partial_flows = [0 for i in range(len(comp_flows[0]))]

    for flow_num in range(len(flows)):
        for comp_num in range(len(comp_flows[0])):
            comp_partial_flows[comp_num] += (flows[flow_num] *
comp_flows[flow_num][comp_num])

    s = sum(flows)

    return [comp_part_flow / s for comp_part_flow in comp_partial_flows]
```

```
comp_flows = [[0.1, 0.1, 0.1, 0.4, 0.2, 0.05, 0.03, 0.02],
              [0.1, 0.2, 0.1, 0.3, 0.1, 0.15, 0.03, 0.02],
              [0.1, 0.1, 0.15, 0.35, 0.1, 0.05, 0.08, 0.07]]

flows = [200, 250, 120]

mix_comp = calc_mix_comp(comp_flows, flows)
```

```

print('Массовые доли компонентов в смеси:')
for comp in mix_comp:
    print(f'{comp:.4f}')
print(f'Сумма = {sum(mix_comp):.4f}')

```

Ответ:

Массовые доли компонентов в смеси:

0.1000

0.1439

0.1105

0.3456

0.1351

0.0939

0.0405

0.0305

Сумма = 1.0000

ЗАДАНИЕ 3

Абсолютная плотность газов и паров [кг/м³] вычисляется по формуле:

$$\rho = \frac{M}{22.4} \cdot \frac{T_0 \cdot P}{T \cdot P_0}$$

где M – молярная масса газа или пара, [кг/кмоль];

$T_0 = 273.15$ – нормальная температура, [K];

T – температура, при которой определяется плотность, [K];

$P_0 = 101325$ – нормальное давление, [Па];

P – давление, при котором определяется плотность, [Па].

Необходимо реализовать функцию для определения плотности метана (CH_4) при $P = 200$ кПа и температуре $T \in [200; 500]$ с шагом $h = 50$ [K].

РЕШЕНИЕ 3

Программная реализация:

```
def calc_rho_in_t_range(
    gas_mole_mass: float,
    press: float,
    t_range: list[float],
    t_step: float
) -> float:

    tn = 273.15
    pn = 101325
    rho_list = []
    t_calc_range = []
    dens = []

    t_calc_range.append(t_range[0])
    number_intervals = int((t_range[1] - t_range[0]) / t_step)
    for i in range(number_intervals):
        t_calc_range.append(t_step + t_calc_range[-1])
    if t_calc_range[-1] < t_range[1]:
        t_calc_range.append(t_range[1])
    for t in t_calc_range:
        rho = gas_mole_mass / 22.4 * tn * press / pn / t
        dens.append(rho)

    return t_calc_range, dens, number_intervals
```

```
methane_mm = 16
press = 200 * 1000
t_range = [200, 500]
h = 50

t_points, dens_points, nn = calc_rho_in_t_range(methane_mm, press, t_range, h)
for i in range(len(t_points)):
    print(f'T = {t_points[i]:.1f} K \t Плотность = {dens_points[i]:.3f} кг/м³')
```

Ответ:

$T = 200.0 \text{ K}$	Плотность = 1.926 кг/м^3
$T = 250.0 \text{ K}$	Плотность = 1.540 кг/м^3
$T = 300.0 \text{ K}$	Плотность = 1.284 кг/м^3
$T = 350.0 \text{ K}$	Плотность = 1.100 кг/м^3
$T = 400.0 \text{ K}$	Плотность = 0.963 кг/м^3
$T = 450.0 \text{ K}$	Плотность = 0.856 кг/м^3
$T = 500.0 \text{ K}$	Плотность = 0.770 кг/м^3

ЗАДАНИЕ 4

Коэффициент сжимаемости учитывает отклонение реального газа от уравнения состояния идеального газа. При точных расчетах коэффициент сжимаемости определяют по формуле:

$$z = 1 + \frac{P_r}{T_r} \cdot \left(0.144 + 0.073 \cdot \omega - \frac{0.33 - 0.46 \cdot \omega}{T_r} - \frac{0.138 + 0.5 \cdot \omega}{T_r^2} - \frac{0.012 + 0.097 \cdot \omega}{T_r^3} - \frac{0.0073 \cdot \omega}{T_r^8} \right)$$

где ω – ацентрический фактор, вычисляемый по уравнению:

$$\omega = \frac{3}{7} \cdot \left(\frac{\lg(P_r) - 5}{\frac{T}{T_r \cdot T_b} - 1} \right) - 1$$

где T_r – приведенная температура: $T_r = \frac{T}{T_c}$

P_r – приведенное давление;

$T_b = 272.65$ – температура кипения, [K];

$T_c = 425.15$, [K].

Необходимо реализовать функции для расчета фактора сжимаемости z при $T \in [200; 400]$ с шагом $h = 25$ [K].

РЕШЕНИЕ 4

Программная реализация:

```
from math import *

def calc_z(
    t_range: list[float],
    t_step: float
) -> float:

    pr = 0.2634
    tb = 272.65
    tc = 425.15

    t_calc_range = []
    z_list = []

    t_calc_range.append(t_range[0])
    number_intervals = int((t_range[1] - t_range[0]) / t_step)
    for i in range(number_intervals):
        t_calc_range.append(t_step + t_calc_range[-1])
    if t_calc_range[-1] < t_range[1]:
        t_calc_range.append(t_range[1])

    for t in t_calc_range:
        tr = t / tc
        w = 3 / 7 * (log10(pr) - 5) / (t / tr / tb - 1) - 1
        z = 1 + pr / tr * (0.144 + 0.073 * w - (0.33 - 0.46 * w) / tr - (0.138 +
0.5 * w) / tr**2 -
                                (0.012 + 0.097 * w) / tr**3 - 0.0073 * w / tr**8)
        z_list.append(z)

    return t_calc_range, z_list

t_range = [200, 400]
h = 25
t_points, z_points= calc_z(t_range, h)

for i in range(len(t_points)):
    print(f'T = {t_points[i]:.2f} K\tz = {z_points[i]:.3f}')
```

Ответ:

$$T = 200.00 \text{ K} \quad z = 15.586$$

$$T = 225.00 \text{ K} \quad z = 7.522$$

$$T = 250.00 \text{ K} \quad z = 4.338$$

$$T = 275.00 \text{ K} \quad z = 2.863$$

$$T = 300.00 \text{ K} \quad z = 2.090$$

$$T = 325.00 \text{ K} \quad z = 1.647$$

$$T = 350.00 \text{ K} \quad z = 1.377$$

$$T = 375.00 \text{ K} \quad z = 1.203$$

$$T = 400.00 \text{ K} \quad z = 1.088$$

Проверка:

$$\begin{aligned}
 Pr &:= 0.2634 \\
 T &:= 200 \\
 T_c &:= 425.15 \\
 T_b &:= 272.65 \\
 Tr &:= \frac{T}{T_c} = 0.47 \\
 w &:= \frac{3}{7} \cdot \left(\frac{\log(Pr) - 5}{\frac{T}{Tr \cdot T_b} - 1} \right) - 1 = -5.275 \\
 z &:= 1 + \frac{Pr}{Tr} \cdot \left(0.144 + 0.073 \cdot w - \frac{0.33 - 0.46 \cdot w}{Tr} - \frac{0.138 + 0.5 \cdot w}{Tr^2} - \frac{0.012 + 0.097 \cdot w}{Tr^3} - \frac{0.0073 \cdot w}{Tr^8} \right) = 15.586
 \end{aligned}$$

ЗАДАНИЕ 5

Реализуйте функцию, возвращающую словарь, в котором ключами будут имена C_1 - C_5 , а значениями другой словарь, содержащий молекулярную массу, температуру и плотность соответствующих алканов. Общая формула для алканов: C_nH_{2n+2} .

1. Температуру кипения можно определить по следующей формуле:

$$T_b = 1090 - \exp\left(6.9955 - 0.11193 \cdot N_c^{\frac{2}{3}}\right)$$

где N_C – число атом углерода в молекуле алкана.

2. Формула для вычисления плотности:

$$\rho = 1.07 - \exp(3.56073 - 2.93886 \cdot MW^{0.1})$$

где MW – молекулярная масса алкана.

РЕШЕНИЕ 5

Программная реализация:

```
def make_dict(
    names: list[str]
) -> dict[dict[list]]:

    c_atom_numbers = [int(name[1:]) for name in names]
    mms = [c_atom_number * 14 + 2 for c_atom_number in c_atom_numbers]
    boiling_temps = [1090 - exp(6.9955 - 0.11193 * c_atom_number) for c_atom_number
in c_atom_numbers]
    rhos = [1.07 - exp(3.56073 - 2.93886 * mm**0.1) for mm in mms]

    res_dict = {names[i]: {'MW': mms[i], 'Tb': boiling_temps[i],
        'rho': rhos[i]} for i in range(len(names))}

    return res_dict
```

```
alkanes = ['C1', 'C2', 'C3', 'C4', 'C5', 'C6', 'C7']
res_dict = make_dict(alkanes)
res_dict
```

Ответ:

```
{'C1': {'MW': 16, 'Tb': 113.89515870858975, 'rho': 0.3417559246621025},
'C2': {'MW': 30, 'Tb': 217.2579821220337, 'rho': 0.50374382536956},
'C3': {'MW': 44, 'Tb': 309.67538982596864, 'rho': 0.5880575794529687},
'C4': {'MW': 58, 'Tb': 392.30642644686316, 'rho': 0.6426337863150681},
'C5': {'MW': 72, 'Tb': 466.18740171633976, 'rho': 0.6819127875116566},
'C6': {'MW': 86, 'Tb': 532.244887256693, 'rho': 0.7120372497072657},
'C7': {'MW': 100, 'Tb': 591.3073370957468, 'rho': 0.7361452322267332}}
```