**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 |
| **PYTHON 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:

где  – мольная доля *i*‑го компонента;

 – массовая доля *i*‑го компонента;

 – молярная масса *i*‑го компонента;

 – число компонента в системе;

 – индекс компонента в системе.

1. Плонтость потока:

где  – плоность потока;

 – массовая доля *i*‑го помпонента;

плотность *i*‑го компонента;

 – число компонента в системе;

 – индекс компонента в системе.

1. Средняя молекурялная масса потока:

где  – средняя молекулярная масса потока;

 – массовая доля *i*‑го помпонента;

 – молярная масса *i*‑го компонента;

 – число компонента в системе;

 – индекс компонента в системе.

# Решение 1

**Програмная 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 |  |  |  |  |  |  |  |  |
| 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 смесевого потока можно найти следующим образом:

где  – массовая доля *i*‑го компонента в смесевом потоке;

 – массовая доля *i*‑го компонента в *j*‑ом потоке;

 – массовый расход *j*‑го потока;

 – индекс потока;

 – индекс компонента в системе;

 – число потоков, подаваемых на смешение;

# Решение 2

**Програмная 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] вычисляет по формуле:

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

 – нормальная температура, [К];

 – температура, при которой определяется плотность, [К];

 – нормальное давление, [Па];

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

Необходимо реализовать функцию для определения плотности метана (*CH4*) при *P* = 200 кПа и температуре *T ∈*[200; 500] с шагом *h* = 50 [K].

# Решение 3

**Програмная 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\tПлотность = {dens\_points[i]:.3f} кг/м3')

**Ответ:**

T = 200.0 K Плотность = 1.926 кг/м3

T = 250.0 K Плотность = 1.540 кг/м3

T = 300.0 K Плотность = 1.284 кг/м3

T = 350.0 K Плотность = 1.100 кг/м3

T = 400.0 K Плотность = 0.963 кг/м3

T = 450.0 K Плотность = 0.856 кг/м3

T = 500.0 K Плотность = 0.770 кг/м3

# Задание 4

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

где  – ацентрический фактор, вычисляемый по уравнению:

где  – приведення температура:

 – приведенное давление;

 – температура кипения, [К];

, [К].

Необходимо реализовать функции для рассчета фактора сжимаемости 𝑧 при *T*∈ [200; 400] с шагом *h* = 25 [K].

# Решение 4

**Програмная 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 K z = 15.586

T = 225.00 K z = 7.522

T = 250.00 K z = 4.338

T = 275.00 K z = 2.863

T = 300.00 K z = 2.090

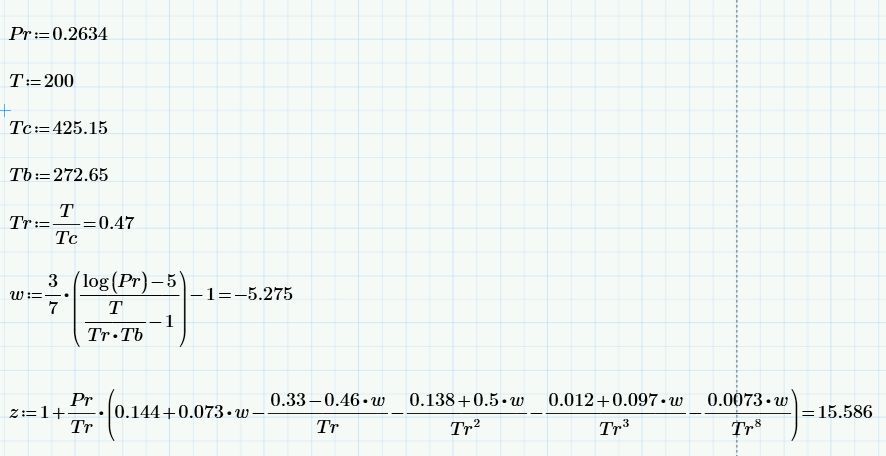
T = 325.00 K z = 1.647

T = 350.00 K z = 1.377

T = 375.00 K z = 1.203

T = 400.00 K z = 1.088

**Проверка:**



# Задание 5

Реализуйте функцию, возвращающую словарь, в котором ключами будут имена *C1‑C5*, а значениями другой словарь, содержащий молекулярную массу, темперутару и плотность соответствующих алканов. Общая формула для алканов: *CnH2n+2*.

1. Температуру кипения можно определить по следующей формуле:

где  – число атом углерода в молекуле алкана.

1. Формула для вычислени плотности:

где  – молекулярная масса алкана.

# Решение 5

**Програмная 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}}