```
#1 a) Use Fuller’s method to estimate the diffusivity of benzene in nitrogen at 311.3 K and 1
atm. Compare your estimate to the value reported in Table 3.2 in the text. Report the %
difference (% difference = (estimated value – table value)/table value * 100%) between your
estimated value and the reported value.
'''
print('============================')
print('Question #1a')
d_AB = HomeworkTwo.FullerDiffusivity(1, # pressure in atmospheres
                                    311.3, # temperature in kelvin
                                    78.114, # molecular weight species A
                                    28.013, # molecular weight species B
                                    15.9 * 6 + 2.31 * 6 - 18.3, # diffusion volume species A
                                    18.5, # diffusion volume species B
                                    'Benzene', # name of species A
                                    'Nitrogen' # name of species B
                                )

HomeworkTwo.percentDifference(d_AB, 0.102)
```

```
================================
Question #1a
mole_weight_ratios_Benzene_Nitrogen: 41.24
diffusivity_Benzene_Nitrogen: 0.100718 cm²/s
Percent difference is -1.25730061906837 [x, x_ref] [0.10071755336855, 0.102]
```

```
'''
b) Use Fuller’s method to estimate the diffusivity of cyclohexane in hydrogen at 288.6 and 1
atm. Compare your estimate to the value reported in Table 3.2 in the text. Report the %
difference (% difference = (estimated value – table value)/table value * 100%) between your
estimated value and the reported value. % difference.
'''
print('============================')
print('Question #1b')

d_AB = HomeworkTwo.FullerDiffusivity(1, # pressure in atmospheres
                                    288.6, # temperature in kelvin
                                    84.15, # molecular weight species A
                                    2.016, # molecular weight species B
                                    15.9 * 6 + 2.31 * 12, # diffusion volume species A
                                    6.2, # diffusion volume species B
                                    'Cyclohexane', # name of species A
                                    'Hydrogen' # name of species B
                                )

HomeworkTwo.percentDifference(d_AB, 0.319)
```

```
================================
Question #1b
mole_weight_ratios_Cyclohexane_Hydrogen: 3.94
diffusivity_Cyclohexane_Hydrogen: 0.313833 cm²/s
Percent difference is -1.61964037251998 [x, x_ref] [0.31383334721166, 0.319]
```

```python
'''
c) Use the table value of the diffusivity of ammonia in hydrogen at 298K and 1 atm to estimate
the diffusivity at 533K. Report the % difference (% difference = (estimated value □ table
value)/table value * 100%) between your estimated value and the reported value. % difference.
'''
print('==============================')
print('Question #1c')

d_AB_1, t_1 = .783, 298
d_AB_2_table, t_2 = 2.149, 533

d_AB = HomeworkTwo.SimplifiedFullerDiffusivity(d_AB_1, t_1, t_2)

HomeworkTwo.percentDifference(d_AB, d_AB_2_table)
```

```
==============================
Question #1c
simple_fuller_diffusivity: 2.165987 cm²/s
Percent difference is 0.79047692684667 [x, x_ref] [2.16598734915793, 2.149]
```

```python
'''
#2 a) Estimate the liquid diffusivity of acetic acid (solute) in a dilute solution of acetone at 288K
using the Wilke-Chang method and the Tyn and Calus method. Compare your estimate to the
value reported in Table 3.4 in the text. Report the % difference (% difference = (estimated value
- table value)/table value * 100%) between your estimated value and the reported value.
'''

# A = acetic acid (t_b 118 C), B = acetone (t_b 56.29 C)
t, mu_b, mVol_A_normal = 288, 0.2379, 87.32

print('=============================')
print('Question #2a WilkeChangDiffusivity')

d_AB = HomeworkTwo.WilkeChangDiffusivity(t, # temperature in K
                                          58.080, # molecular weight of B in g/mol
                                          mu_b, # viscosity of B in cp
                                          mVol_A_normal, # molar volume of A at normal boiling in cm**3/mol
                                          1.0 # association factor ??
                                          )

d_AB_Table = 2.92 * 10**-5
HomeworkTwo.percentDifference(d_AB, d_AB_Table)

print('=============================')
print('Question #2a TynCalusDiffusivity')

d_AB = HomeworkTwo.TynCalusDiffusivity(t, # abs temperature in kelvin
                                        mVol_A_normal, # molar volume of A at normal boiling point in cm**3/mol
                                        59.47, # molar volume of B at normal boiling point in cm**3/mol
                                        17.98, # surface tension of A at temp, t in dyn/cm
                                        19.22, # surface tension of B at temp, t in dyn/cm
                                        mu_b, # viscosity of B in cp
                                        'organic_acid', # [water, organic_acid, nonpolar_into_monohydroxy_alcohols]
                                        )

HomeworkTwo.percentDifference(d_AB, d_AB_Table)
```

```
=============================
Question #2a WilkeChangDiffusivity
WilkeChangDiffusivity: 4.7e-05
Percent difference is 60.02652760053498 [x, x_ref] [4.672774606e-05, 2.92e-05]
=============================
Question #2a TynCalusDiffusivity
para_A: 359.6178 para_B: 124.5192
TynCalusDiffusivity: 3.5e-05
Percent difference is 18.67567520503005 [x, x_ref] [3.465329716e-05, 2.92e-05]
```

```python
'''
b) Estimate the liquid diffusivity of water (solute) in a dilute solution of ethanol (solvent) at
298K using the Wilke-Chang method and the Tyn and Calus method. Compare your estimate to
the value reported in Table 3.4 in the text. Report the % difference (% difference = (estimated
value – table value)/table value * 100%) between your estimated value and the reported value.
'''
print('============================')
print('Question #2b WilkeChangDiffusivity')

# A = water (t_b 100 C), B = ethanol (t_b 78.29 C)
t, mu_b, mVol_A_normal = 298, 0.4488, 18.93


d_AB = HomeworkTwo.WilkeChangDiffusivity(t, # temperature in K
                                          46.08, # molecular weight of B in g/mol
                                          mu_b, # viscosity of B in cp
                                          mVol_A_normal, # molar volume of A at normal boiling in cm**3/mol
                                          1.5 # association factor
                                         )
d_AB_Table = 1.26 * 10**-5
HomeworkTwo.percentDifference(d_AB, d_AB_Table)

print('============================')
print('Question #2b TynCalusDiffusivity')
d_AB = HomeworkTwo.TynCalusDiffusivity(t, # abs temperature in kelvin
                                        mVol_A_normal, # molar volume of A at normal boiling point in cm**3/mol
                                        62.62, # molar volume of B at normal boiling point in cm**3/mol
                                        57.09, # surface tension of A at temp, t in dyn/cm
                                        17.57, # surface tension of B at temp, t in dyn/cm
                                        mu_b, # viscosity of B in cp
                                        'water' # [water, organic_acid, nonpolar_into_monohydroxy_alcohols]
                                       )

HomeworkTwo.percentDifference(d_AB, d_AB_Table)
```

```
================================
Question #2b WilkeChangDiffusivity
WilkeChangDiffusivity: 7×10⁻⁵ cm²/s
Percent difference is 455.3107575975571 [x, x_ref] [6.996915546e-05, 1.26e-05]
================================
Question #2b TynCalusDiffusivity
para_A: 104.0688 para_B: 128.2053
TynCalusDiffusivity: 3.1×10⁻⁵ cm²/s
Percent difference is 146.10549277887935 [x, x_ref] [3.100929209e-05, 1.26e-05]
```

BASE FUNCTIONS

```python
@staticmethod
def percentDifference(x:float, x2:float):
    value = ((x - x2)/x2)*100
    print(f"Percent difference is {round(value,14)} [x, x_ref] [{round(x,14)}, {round(x2,14)}]")
    return value

@staticmethod
def FullerDiffusivity(
                    p: float, # pressure in atmospheres
                    t: float, # temperature in kelvin
                    molW1: float, # molecular weight species A
                    molW2: float, # molecular weight species B
                    dVol1: float, # diffusion volume species A
                    dVol2: float, # diffusion volume species B
                    specA: str = 'A', # name of species A
                    specB: str = 'B', # name of species B
                    ):

  m_AB = 2 / ((1 / molW1) + (1 / molW2))
  d_AB = (0.00143 * t**1.75) / (p * m_AB**(1/2) * (dVol1**(1/3) + dVol2**(1/3))**2 )

  print(f'mole_weight_ratios_{specA}_{specB}: {round(m_AB,2)}'.lower())
  print(f'diffusivity_{specA}_{specB}: {round(d_AB,6)}'.lower())

  return d_AB

@staticmethod
def SimplifiedFullerDiffusivity(
                    d_AB_1: float,
                    t_1: float,
                    t_2: float,
                    ):
  d_AB = d_AB_1 * (t_2 / t_1)**(1.75)

  print(f'simple_fuller_diffusivity: {round(d_AB,6)}')
  return d_AB

@staticmethod
def WilkeChangDiffusivity( # A:solute, B:solvent
    t: float, # temperature in K
    mW_B: float, # molecular weight of B in g/mol
    mu_B: float, # viscosity of B in cp
    mVol_A_normal: float, # molar volume of A at normal boiling in cm**3/mol
    aF: float = 1.5 # association factor {2.6: water, 1.9: methanol, 1.5: ethanol, 1.0: unassociated}
  ):

  d_AB = (7.4 * 10**-8 * (aF*mW_B)**(1/2) * t) / (mu_B * mVol_A_normal**0.6)

  print(f'WilkeChangDiffusivity: {round(d_AB,6)}')
  return d_AB
```

```python
@staticmethod
def TynCalusDiffusivity( # A:solute, B:solvent
        t: float, # abs temperature in kelvin
        mVol_A_normal: float, # molar volume at normal boiling point in cm**3/mol
        mVol_B_normal: float, # molar volume at normal boiling point in cm**3/mol
        sTension_A_t: float, # surface tension of A at temp, t in dyn/cm
        sTension_B_t: float, # surface tension of B at temp, t in dyn/cm
        mu_B: float, # viscosity of B in cp
        solute: str # [water, organicAcid, nonpolar_into_monohydroxy_alcohols]
    ):

    if mu_B > 30:
        raise Exception('mu_B > 30 cp')

    def Parachor(mVol_at_t: float, sTension_at_t: float):
        return (mVol_at_t * sTension_at_t**(1/4))

    if solute in ['water', 'organic_acid']:
        #2. If solute is water, double mVol_A and para_A
        #3. If solute is organic acid and solvent is other than water, methanol, or butanol, double mVol_A and para_A

        para_A = Parachor(mVol_A_normal, sTension_A_t) * 2
        para_B = Parachor(mVol_B_normal, sTension_B_t)

        d_AB = (
                8.93 * 10**-8 *
                (mVol_A_normal * 2 / mVol_B_normal**2)**(1/6) *
                (para_B / para_A)**.6 * t / mu_B
               )

    elif solute == 'nonpolar_into_monohydroxy_alcohols':
        # For nonpolar solutes diffusing into monohydroxy alcohols, multiply mVol_B and para_B by 8*mu_B

        para_A = Parachor(mVol_A_normal, sTension_A_t)
        para_B = Parachor(mVol_B_normal, sTension_B_t) * 8 * mu_B

        d_AB = (8.93 * 10**-8 *
                (mVol_A_normal / (mVol_B_normal**2 * 8 * mu_B))**(1/6) *
                (para_B / para_A)**.6 * t / mu_B
               )

    else:
        para_A = Parachor(mVol_A_normal, sTension_A_t)
        para_B = Parachor(mVol_B_normal, sTension_B_t)

        d_AB = (
                8.93 * 10**-8 *
                (mVol_A_normal / mVol_B_normal**2)**(1/6) *
                (Parachor(mVol_A_normal, sTension_A_t) / Parachor(mVol_B_normal, sTension_B_t)) *
                (para_B / para_A)**.6 * t / mu_B
               )

    print(f'para_A: {round(para_A, 4)}', f'para_B: {round(para_B, 4)}')
    print(f'TynCalusDiffusivity: {round(d_AB,6)}')
    return d_AB
```