Quiz 9 Number Density and Neutrons

March 27, 2023

[23]:
```
#This assignment was completed with the IAEA's values for the isotopes:
#https://www-nds.iaea.org/relnsd/vcharthtml/VChartHTML.html
import pandas as pd
import urllib.request
from math import *

def lc_read_csv(url):
    req = urllib.request.Request(url)
    req.add_header('User-Agent', 'Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:77.
 ↪0) Gecko/20100101 Firefox/77.0')
    return pd.read_csv(urllib.request.urlopen(req))
# the service URL
livechart = "https://nds.iaea.org/relnsd/v0/data?"
lc_iso = livechart + "fields=ground_states"

def iso_molar(iso): #accepts string of Z number followed by atomic symbol
    iso_info = lc_read_csv(lc_iso + f"&nuclides={iso}")
    mu_iso_molar = iso_info["atomic_mass"][0]
    iso_molar = mu_iso_molar/10**6
    return iso_molar #returns molar mass in amu

def iso_bun(iso): #accepts string of Z number followed by atomic symbol
    iso_info = lc_read_csv(lc_iso + f"&nuclides={iso}")
    iso_abund = iso_info["abundance"][0]/100
    return iso_abund #returns molar mass in amu

def thalf(iso): #accepts string of Z number followed by atomic symbol
    iso_info = lc_read_csv(lc_iso + f"&nuclides={iso}")["half_life_sec"][0]
    thalf = iso_info
    return thalf #returns half life in seconds

def u_mev(u):
    MeV = u * (9.31494028*100)
    return MeV

def decay(init,duration,thalf): #Accepts initial quantity of material, the␣
 ↪material's half life in seconds, and the duration it decays in seconds
```

```
    dconst = log(2)/thalf
    final = init*exp(-dconst*duration)
    return final

avo = 6.02214*10**23 #atoms/mol ; avogadros numebr
c = 299792458 # m/s ; speed of light
```

[18]:
```
u_235_thalf = thalf("235U")
u_235_amu = iso_molar("235U")
u_235_bun = iso_bun("235U")

u_238_amu = iso_molar("238U")

u_amu = 238.02891
o_amu = 15.9994
```

[74]:
```
#Problem 1:

#Given values and conversions:
tonnes_uo2 = 2.5
kg_uo2 = 1000*tonnes_uo2 #Metric tonnes to kg equal tonnes*1000
g_uo2 = kg_uo2*1000
u_init = u_amu
duration = 1 #week
duration = duration*7*24*3600 #seconds
```

[86]:
```
#Problem 1:

#Calculations:

'''
This question does not say whether the final product is Uranium oxide
or if it's pure uranium, no longer accounting for mass of the oxygen.
It also does not say whether the percentage of uranium is mass % or atom %.
Due to this I will be giving all those results.
'''
molar_uo2 = u_init + 2*o_amu
moles_uo2 = g_uo2/molar_uo2
mole_o = 2*moles_uo2
moles_u = moles_uo2
atoms_u = moles_u*avo

atoms_u_235 = atoms_u*u_235_bun
atoms_u_235 = decay(atoms_u_235, duration, u_235_thalf) #current atoms 1 week␣
 ↪after initial

#mass % u_235 in just a uranium sample
```

```
moles_u_235 = atoms_u_235/avo
g_u_235 = moles_u_235 *u_235_amu
massperc_enr_u_g = g_u_235/.95
massperc_enr_u_kg = massperc_enr_u_g/1000
print(f"Problem 1:\n {massperc_enr_u_kg} kilograms enriched Uranium\n --Using␣
↪mass percent")

#mass % u_235 in a uranium oxide sample
#ASSUMPTION: The enriched sample contains no U-234.
#--If it did I would have to invent a ratio of U-234 to U-238 not stated by the␣
↪problem
g_u_238_m_enr = (massperc_enr_u_g*.05)
moles_u_238_m_enr = g_u_238_m_enr/u_238_amu
moles_u = moles_u_235 + moles_u_238_m_enr
moles_o_m_enr = moles_u*2
g_o_m_enr = moles_o_m_enr*o_amu #grams of oxygen in mass enriched sample
g_uo2_m_enr = g_o_m_enr + massperc_enr_u_g #grams total in enriched sample
kg_uo2_m_enr = g_uo2_m_enr/1000
print(f"\n {kg_uo2_m_enr} kilograms enriched Uranium oxide\n --Using mass␣
↪percent")

#atom % u_235 in just a uranium sample
atperc_enr_u_atoms = atoms_u_235/.95 #number of atoms U total in an enriched␣
↪sample
atperc_enr_u_238_atoms = .05*atperc_enr_u_atoms
atperc_enr_u_238_moles = atperc_enr_u_238_atoms/avo
atperc_enr_u_238_g = atperc_enr_u_238_moles*u_238_amu
atperc_enr_u_235_moles = atoms_u_235/avo
atperc_enr_u_235_g = atperc_enr_u_235_moles*u_235_amu
atperc_enr_u_g = atperc_enr_u_235_g + atperc_enr_u_238_g
atperc_enr_u_kg = atperc_enr_u_g/1000
print(f"\n {atperc_enr_u_kg} kilograms enriched Uranium\n --Using atom percent")

#atom % u_235 in a uranium oxide sample
atperc_u_moles = atperc_enr_u_238_moles + atperc_enr_u_235_moles
moles_o_at_enr = atperc_u_moles*2
g_o_at_enr = moles_o_at_enr*o_amu
g_uo2_at_enr = atperc_enr_u_g + g_o_at_enr
kg_uo2_at_enr = g_uo2_at_enr/1000
print(f"\n {kg_uo2_at_enr} kilograms enriched Uranium oxide\n --Using atom␣
↪percent")
```

```
Problem 1:
 16.501780679071366 kilograms enriched Uranium
 --Using mass percent

 18.746908632719556 kilograms enriched Uranium oxide
```

```
--Using mass percent

16.512335838943304 kilograms enriched Uranium
--Using atom percent

18.758882617767657 kilograms enriched Uranium oxide
--Using atom percent
```