```
"""
    File problem_2_transfers.py created on 29/11/2021 by sebrimmer at 15:04:39

    Current Directory: HW3

    Compute the total ΔV required to send a spacecraft a from a geocentric circular
    orbit of 7000 km radius to geocentric circular orbit of 105000 km radius using
    a Hohmann transfer. Repeat the ΔV computation using a bi-elliptic transfer where
    apogee of the intermediate orbit is 210000 km. Which approach requires less ΔV?
    At what value, to the nearest km, of apogee of the intermediate orbit does the
    other approach require the least ΔV?

"""
import numpy as np
from math import pi, sqrt, cos, sin, acos, asin, tan, atan


def main():

    mu_earth = 398_600
    r_1 = 7_000
    r_2 = 105_000

    bi_elip_intermed_apogee = 210_000

    # Hohmann transfer semi-major axis
    a_h = (r_1 + r_2) / 2

    # Assuming circular orbits, the velocities at periapse/apoapse are:
    v_periapse = sqrt(mu_earth * (2 / r_1 - 1 / a_h))
    v_apoapse = sqrt(mu_earth * (2 / r_2 - 1 / a_h))

```

```
33      orbit_1_circ_velocity = sqrt(mu_earth / r_1)
34      orbit_2_circ_velocity = sqrt(mu_earth / r_2)
35
36      delta_v_a = v_periapse - orbit_1_circ_velocity
37      delta_v_b = orbit_2_circ_velocity - v_apoapse
38      delta_v_total_hohmann = delta_v_a + delta_v_b
39
40      # Now performing calculations for bi-elliptic
    transfer
41      a_h_be_first = (r_1 + bi_elip_intermed_apogee
    ) / 2  # b-e semi-major axis for first leg
42      a_h_be_apogee_first = a_h_be_first * 2 - r_1  #
    b-e apogee, calculated from semi-major axis and r1
43
44      # Velocity at perigee on the first leg (i.e.
    first Hohmann transfer) for the Bielliptic transfer
     (km/s)
45      v_be_periapse = sqrt(mu_earth * (2 / r_1 - 1 /
    a_h_be_first))
46
47      # Velocity at apogee on the first leg (i.e.
    first Hohmann transfer) for the Bielliptic transfer
     (km/s)
48      v_be_apoapse = sqrt(mu_earth * (2 /
    a_h_be_apogee_first - 1 / a_h_be_first))
49
50      # Delta V to get from the circular 7000 km orbit
     onto the first leg (i.e. first Hohmann transfer)
51      # for the Bielliptic transfer:
52      # delta v = what we want - what we have
53      delta_v_a_be_to_first_leg = v_be_periapse -
    orbit_1_circ_velocity
54
55      # Semimajor axis for transfer from the 210000 km
     orbit to the 105000 km orbit (km)
56      a_h_be_second = (a_h_be_apogee_first + r_2) / 2
57
58      # Velocity at apogee on the second leg (i.e.
    second Hohmann transfer) for the Bielliptic transfer
     (km/s)
59      v_be_apogee_second = sqrt(mu_earth * (2 /
```

```
59 a_h_be_apogee_first - 1 / a_h_be_second))
60     v_be_perigee_second = sqrt(mu_earth * ((2 / r_2
   ) - (1 / a_h_be_second)))
61
62     # Delta V to transfer from the first leg (i.e.
   first Hohmann transfer)
63     # to the second leg (i.e. second Hohmann
   transfer) for the Bielliptic transfer (km/s)
64     delta_v_b_be_to_second_leg = v_be_apogee_second
    - v_be_apoapse
65
66     # Delta V to get from the second leg (i.e.
   second Hohmann transfer)
67     # onto the circular orbit of radius 105000 km (
   km/s)
68     delta_v_b_be_to_second_orbit =
   v_be_perigee_second - orbit_2_circ_velocity
69
70     # Total DV for the Bielliptic transfer (km/s)
71     # total dv = dv to get onto first transfer
   ellipse
72     #            + dv to get onto second transfer
   ellipse
73     #            + dv to get onto second orbit
74     delta_v_total_bi_elliptic =
   delta_v_a_be_to_first_leg \
75                                    +
   delta_v_b_be_to_second_leg \
76                                    +
   delta_v_b_be_to_second_orbit
77
78     # Radius of intermediate orbit such that the
   other transfer option can be calculated iteratively
   or analytically.
79     # We use the iterative method, and decrease the
   intermediate orbit semi-maj axis until the delta v
   values match
80
81     bi_elip_intermed_apogee = 210_000
82     delta_v_total_bi_elliptic = 4.029   # initial
   delta v for hohmann is 4.046
```

```
 83
 84     while delta_v_total_bi_elliptic <=
    delta_v_total_hohmann:
 85
 86         a_h_be_first_iter = (r_1 +
    bi_elip_intermed_apogee) / 2  # b-e semi-major axis
     for first leg
 87         a_h_be_apogee_first_iter =
    a_h_be_first_iter * 2 - r_1  # b-e apogee,
    calculated from semi-major axis and r1
 88
 89         # Velocity at perigee on the first leg (i.e
    . first Hohmann transfer) for the Bielliptic
    transfer (km/s)
 90         v_be_periapse_iter = sqrt(mu_earth * (2 /
    r_1 - 1 / a_h_be_first_iter))
 91
 92         # Velocity at apogee on the first leg (i.e
    . first Hohmann transfer) for the Bielliptic
    transfer (km/s)
 93         v_be_apoapse_iter = sqrt(mu_earth * (2 /
    a_h_be_apogee_first_iter - 1 / a_h_be_first_iter))
 94
 95         # Delta V to get from the circular 7000 km
    orbit onto the first leg (i.e. first Hohmann
    transfer)
 96         # for the Bielliptic transfer:
 97         # delta v = what we want - what we have
 98         delta_v_a_be_to_first_leg_iter =
    v_be_periapse_iter - orbit_1_circ_velocity
 99
100         # Semimajor axis for transfer from the
    210000 km orbit to the 105000 km orbit (km)
101         a_h_be_second_iter = (
    a_h_be_apogee_first_iter + r_2) / 2
102
103         # Velocity at apogee on the second leg (i.e
    . second Hohmann transfer) for the Bielliptic
    transfer (km/s)
104         v_be_apogee_second_iter = sqrt(mu_earth * (
    2 / a_h_be_apogee_first_iter - 1 /
```

```
104 a_h_be_second_iter))
105         v_be_perigee_second_iter = sqrt(mu_earth
     * ((2 / r_2) - (1 / a_h_be_second_iter)))
106 
107         # Delta V to transfer from the first leg (i
    .e. first Hohmann transfer)
108         # to the second leg (i.e. second Hohmann
    transfer) for the Bielliptic transfer (km/s)
109         delta_v_b_be_to_second_leg_iter =
    v_be_apogee_second_iter - v_be_apoapse_iter
110 
111         # Delta V to get from the second leg (i.e.
    second Hohmann transfer)
112         # onto the circular orbit of radius 105000
    km (km/s)
113         delta_v_b_be_to_second_orbit_iter =
    v_be_perigee_second_iter - orbit_2_circ_velocity
114 
115         # Total DV for the Bielliptic transfer (km/
    s)
116         # total dv = dv to get onto first transfer
    ellipse
117         #             + dv to get onto second
    transfer ellipse
118         #             + dv to get onto second orbit
119         delta_v_total_bi_elliptic =
    delta_v_a_be_to_first_leg_iter \
120                                           +
    delta_v_b_be_to_second_leg_iter \
121                                           +
    delta_v_b_be_to_second_orbit_iter
122 
123         # decrement the bi_elip_intermed_apogee
    variable for the next iteration
124         bi_elip_intermed_apogee -= 1
125 
126     output_string = f"Hohmann transfer semi-major
    axis: {a_h} km\n" \
127                     f"Apoapse Velocity: {v_apoapse:
    .3f} km\n" \
128                     f"Periapse Velocity: {
```

```
v_periapse:.3f} km\n" \
                      f"Orbit 1 circ velocity: {orbit_1_circ_velocity:.3f}\n" \
                      f"Orbit 2 circ velocity: {orbit_2_circ_velocity:.3f}\n" \
                      f"Delta v1 Velocity: {delta_v_a:.3f} km\n" \
                      f"Delta v2 Velocity: {delta_v_b:.3f} km\n" \
                      f"Total dV: {delta_v_total_hohmann:.3f} km\n\n" \
                      f"Hohmann transfer semi-major axis (bi-elliptic): {a_h_be_first} km\n" \
                      f"2.07) Semimajor axis for transfer from the 7000 km orbit to the " \
                      f"210000 km orbit (km): {a_h_be_first:.3f}\n" \
                      f"" \
                      f"2.08) Velocity at perigee on the first leg (i.e. first Hohmann transfer)  \n" \
                      f"      for the Bielliptic transfer (km/s): {v_be_periapse:.3f}\n" \
                      f"2.09) Delta V to get from the circular 7000 km orbit onto the first leg\n" \
                      f"      (i.e. first Hohmann transfer) for the Bielliptic transfer: {delta_v_a_be_to_first_leg:.3f} km/s\n\n" \
                      f"2.10) Velocity at apogee on the first leg (i.e. first Hohmann transfer)\n" \
                      f"      for the Bielliptic transfer (km/s): {v_be_apoapse:.3f} km/s\n\n" \
                      f"2.11) Semimajor axis for transfer from the 210000 km orbit to the 105000 km " \
                      f"orbit: {a_h_be_second} km/s\n\n" \
                      f"2.12) Velocity at apogee on the second leg (i.e. second Hohmann transfer)\n" \
                      f"      for the Bielliptic transfer (km/s): {v_be_apogee_second:.3f} km/s\n\n" \
```

```
                    f"2.13) Delta V to transfer from the first leg (i.e. first Hohmann transfer)\n" \
                    f"      to the second leg (i.e. second Hohmann transfer) for the Bielliptic \n" \
                    f"      transfer: {delta_v_b_be_to_second_leg:.3f} km/s\n\n" \
                    f"2.14) Velocity at perigee on the second leg (i.e. second Hohmann transfer)\n" \
                    f"      for the Bielliptic transfer: {v_be_perigee_second:.3f} km/s\n\n" \
                    f"2.15) Delta V to get from the second leg (i.e. second Hohmann transfer)\n" \
                    f"      onto the circular orbit of radius 105000 km: {delta_v_b_be_to_second_orbit:.3f} km/s\n\n" \
                    f"2.16) Total DV for the Bielliptic transfer: {delta_v_total_bi_elliptic:.3f} km/s\n\n" \
                    f"2.17) Radius of intermediate orbit such that the other transfer option requires " \
                    f"less DV: {bi_elip_intermed_apogee:.3f} km " \

    print(output_string)

    with open('output/problem_2_output.txt', 'w') as output:
        output.write(output_string)

    return 0


if __name__ == '__main__':
    main()

```