#defining a class for the flights

class Flight:

def \_\_init\_\_(self, flight, departure, arrival, maintenance = 0):

self.flight = flight

self.departure = departure

self.arrival = arrival

self.maintenance = maintenance

#defining a function to convert a 24hr format time (an str) into minutes

def convert\_to\_minutes(time\_str):

hours, minutes = map(int, time\_str.split())

return hours \* 60 + minutes

#defining a function to convert minutes into 24hr format time

def convert\_to\_time(minutes):

hours = minutes // 60

minutes = minutes % 60

return f"{hours:02d} {minutes:02d}"

#Greedy Scheduling Algorithm

#defining the airline scheduling algorithm as a function

def airline\_schedule(flights):

#sorting the flights based on their arrival time

flights.sort(key = lambda x: x.arrival)

schedule = list()

#initializing the current time as zero (i.e, midnight)

current\_time = 0

#scheduling algorithm logic

for flight in flights:

if flight.departure >= current\_time + flight.maintenance:

schedule.append(flight)

current\_time = flight.arrival

return schedule

def get\_user\_input():

flights = list()

while True:

flight = input("Enter Flight Number (or enter \"done\" to finish: ")

if flight.lower() == "done":

break

departure\_str = input("Enter Departure time in 24 hour format: ")

departure = convert\_to\_minutes(departure\_str)

arrival\_str = input("Enter Arrival time in 24 hour format: ")

arrival = convert\_to\_minutes(arrival\_str)

maintenance = int(input("Enter Maintenance time in minutes"))

flights.append(Flight(flight, departure, arrival, maintenance))

return flights

#Main code

flights = get\_user\_input()

scheduled\_flights = airline\_schedule(flights)

print("\nScheduled Flights:")

for flight in scheduled\_flights:

print(f"""Flight {flight.flight}: Departure {convert\_to\_time(flight.departure)},

Arrival {convert\_to\_time(flight.arrival)}""")