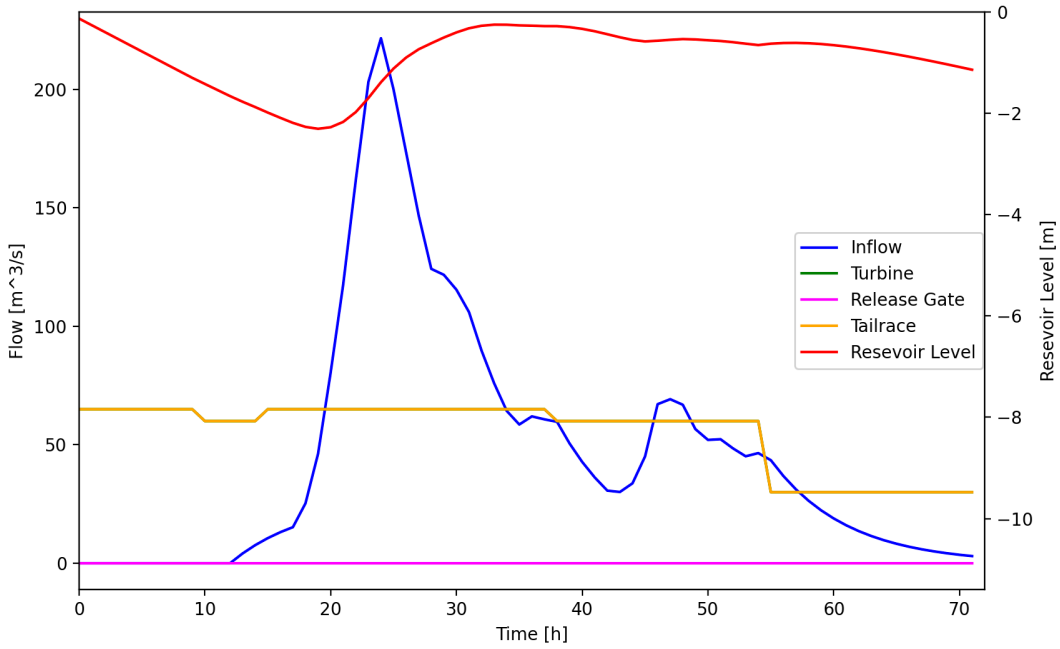
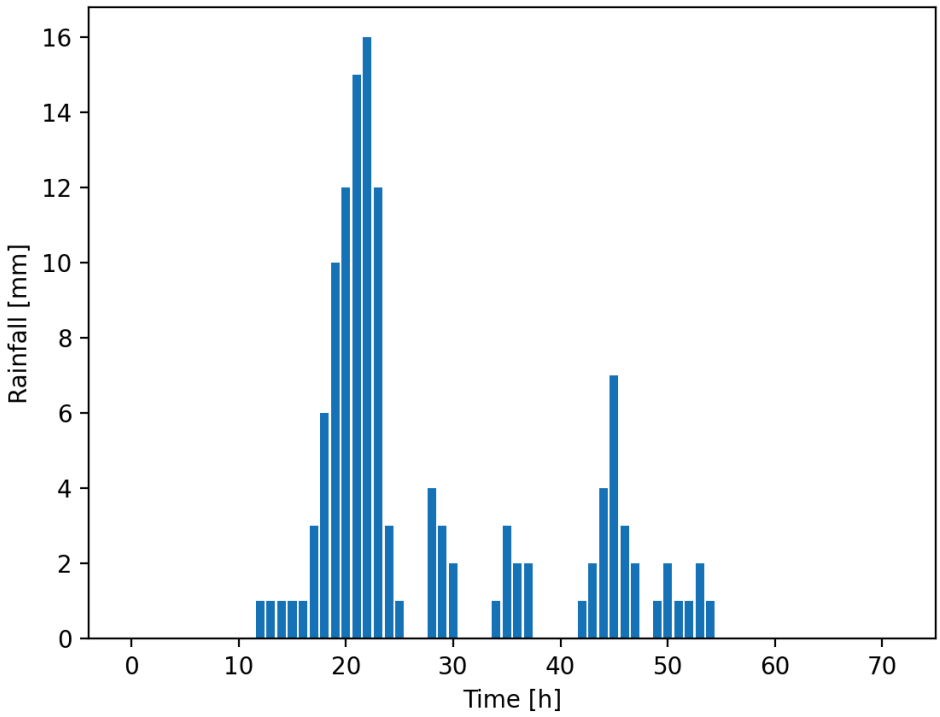
```python
import math
import matplotlib.pyplot as plt
import numpy as np

t_1 = 0 # h
t_2 = 1 # h
Q_1 = 0 # m^3/s
Q_2 = 0 # initializing
R = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 3, 6, 10, 12,
15, 16, 12, 3, 1, 0, 0, 4, 3, 2, 0, 0, 0, 1, 3, 2, 2, 0, 0, 0, 0, 1, 2,
4, 7, 3, 2, 0, 1, 2, 1, 1, 2, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0] # mm/day  - effective rainfall/ water impinging on basin
rainfall = R
A_neg = - 1 / 6 # h^-1

basin_area = 97000000
secs_per_hour = 3600

storage_volume = 20500000 # m
res_surface_area = 1800000 # m
max_depth = 11.389 # m

R = [r / 1000 for r in R] # mm to m conversion
R = [r / secs_per_hour for r in R] # hr to s conversion
R = [r * basin_area for r in R] # water amount converter to volume (m^3/s)


inflow = [0] * 72 # m^3/s

# Equation 3 for inflow
for i in range(0,72):
    inflow[i] = Q_1
    Q_2 = Q_1 * math.exp(A_neg * (t_2 - t_1)) + R[i] * (1 - math.exp(A_neg * (t_2 - t_1)))
    Q_1 = Q_2                    # ^ hr in t_1 & t_2 cancel out with A_neg hr-1 ^
    t_1 += 1
    t_2 += 1

turbine = np.full(72, 65) # turbine on full for whole 72 hours
turbine[10:15] = 60
turbine[38:55] = 60
turbine[55:] = 30
rel_gate = np.zeros(72) # never open release gate
tailrace = np.add(turbine, rel_gate)

# calculate reservoir level
inflow_per_hour = [i * secs_per_hour for i in inflow]
turb_per_hour = [t * secs_per_hour for t in turbine] # convert flow to hourly
rg_per_hour = [rg * secs_per_hour for rg in rel_gate] # convert flow to hourly

res_level = np.zeros(72)
cur_volume = storage_volume
for i, rl in enumerate(res_level):
    cur_volume = cur_volume + inflow_per_hour[i] - turb_per_hour[i] - rg_per_hour[i]
    res_level[i] = cur_volume / res_surface_area - max_depth
    if (rl > 0):
        tailrace[i] += rl

x = list(range(0,72))

fig, host = plt.subplots(1, 1, figsize=(10,6))

par = host.twinx()

p1, = host.plot(x, inflow, label='Inflow', color='blue')
p2, = host.plot(x, turbine, label='Turbine', color='green')
p3, = host.plot(x, rel_gate, label='Release Gate', color='magenta')
p4, = host.plot(x, tailrace, label='Tailrace', color='orange')
p5, = par.plot(x, res_level, label='Resevoir Level', color='red')

host.set_xlim(0,72)
par.set_ylim(-11.4, 0)

host.set_ylabel('Flow [m^3/s]')
host.set_xlabel('Time [h]')
par.set_ylabel('Resevoir Level [m]')

lines = [p1, p2, p3, p4, p5]

host.legend(lines, [l.get_label() for l in lines], loc="center right", borderaxespad=0.1)

plt.subplots_adjust(right=0.85)

plt.show()

plt.bar(x, rainfall)
plt.xlabel('Time [h]')
plt.ylabel('Rainfall [mm]')

plt.show()
```


Flow [m^3/s]
Resevoir Level [m]
Time [h]
Inflow
Turbine
Release Gate
Tailrace
Resevoir Level


Rainfall [mm]
Time [h]