## Leaky bucket algorithm

if \_-vane =: "\_-main\_\_":

Storage = 0

bucket\_Size = int (input (" Einter Man Bucket capacity."))

fin putsize = int (model ("Entor size of Packet doming to I'M)

# cut Flow

entput size = int (impid (" Enter Nullaber of packets goingout)")

I number of Home take is theke of.

fox i in stange (quints):

# Space loft. Size : left = bucket Size - Storage imput size = int (input (" Enter Rumber of Packets 1) (input size <= Sizé\_luft);

Stronge += . Imput Stro

Point ("Buffer Size = Ro & Bucket Size = R1]" found!

Storage, bucker Sizo))

else.

Point ("Packet loss: 203". format (input-aze-Size Sift)

Storage = bucket - SI-Ze

Point (" Buffer Size: Roz Aukot Size: 412"

too mate (Storage, bucketszz) -= output -size