#include<stdio.h>

int main(){

int incoming, outgoing, buck\_capacity, n, store = 0;

printf("Enter bucket capacity, outgoing rate and no of inputs: ");

scanf("%d %d %d", &buck\_capacity, &outgoing, &n);

while (n != 0) { //loop over total number of inputs

printf("Enter the number of incoming packets: ");

scanf("%d", &incoming);

printf("Incoming packet size %d\n", incoming);

if((incoming-outgoing) <= (buck\_capacity-store)) //it is possible to send without dropping

{

int sent = outgoing>=incoming?incoming:outgoing; //if incoming is more than outgoing, total sent will be outgoing rest buff

if(sent < outgoing && store != 0) //if incoming<outgoing, we can add values from the store to be sent till op cap is reached

{

int remaining = outgoing-sent;

while(remaining > 0 && store != 0) //keeps adding one to sent until we run out of op cap or nothing left in buff

{

remaining -= 1;

store -= 1;

sent += 1;

}

}

printf("%d packets sent out\n", sent);

if(outgoing<incoming)

{

store = store + (incoming - outgoing);

}

}

else //packets need to be dropped

{

int dropped = (incoming-outgoing)-(buck\_capacity-store); //excess packets are the ones that after sending cant be accomodated in buff

printf("%d packets dropped\n", dropped);

store = buck\_capacity;

}

printf("%d out of %d space used in the buffer\n", store, buck\_capacity);

n--;

}

}