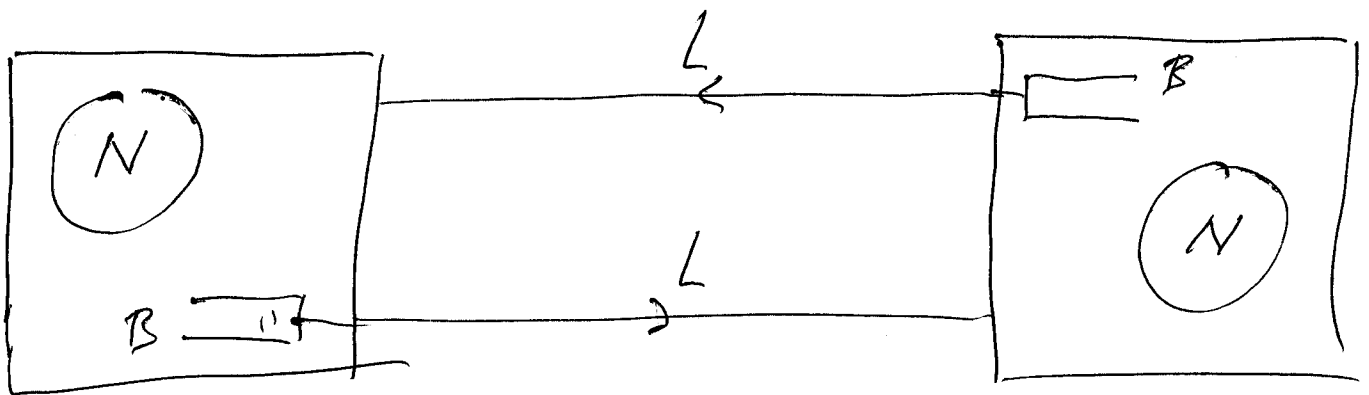


Simulation. Go Back to A

C++

Who did not take CIS 114?

"Object Oriented".



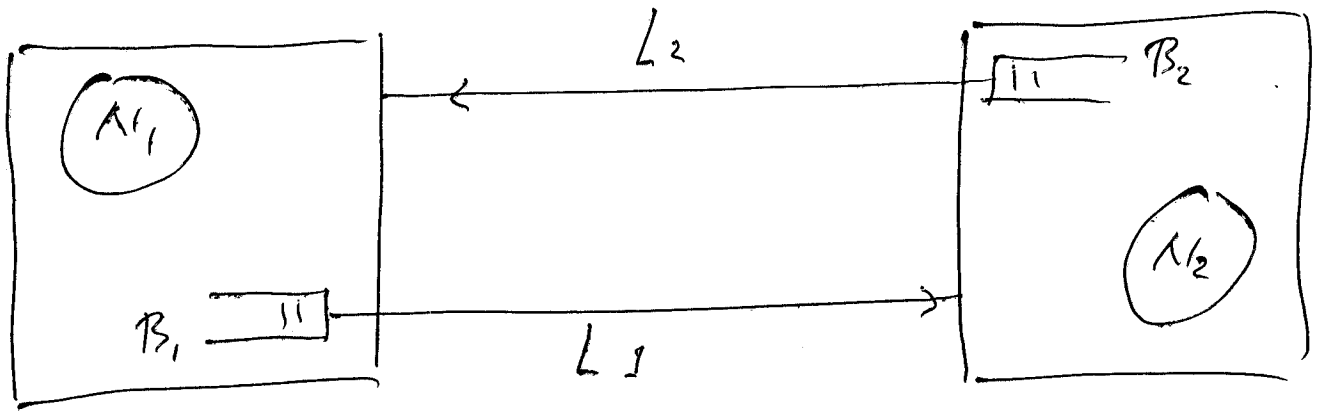
Classes:

Node: Simulates "Software"

Buffer: Simulates "Buffer"
(called: Framebuffer)

Link: Simulates "Link"

Simulation



Two objects of class Node
 Two objects of class Buffer
 Two objects of class Link

Class Frame.

varying number of objects of
 class Frame

(new, delete,
 or construction, destruction).

Normal operation:

N₂ Constructs Delta Frame,
Hands it to B₁.

B₁ enqueues Delta Frame,
at correct time hands it
to L₁

L₁ draws Lottery: $\left\{ \begin{array}{l} P_{Ok} : Ok \\ P_{Lost} : Lost \\ P_{Dam} : Damaged \end{array} \right.$

(Lost: "delete Frame").

if not Lost: in time,

L₁ hands Delta Frame to N₂

N₂ : Sends ACK or NACK,
hands to B₂

B₂ : In time:
Hands to L₂

L2 :

Lottery.
IF Ok :
Hands to N1

N1 :

Receives ACK or NACK,
updates source_acked,
makes more delta_Frames,
(Transmit or Re-transmit)
etc.

Lottery : Use rand() .

In Unix : See man rand

rand() : \approx Unif {~~0, 2~~, 0, 1, ..., RAND_MAX}
(RAND_MAX = $2^{32} - 1$?)

Ms kangiluppi :

myrand()	return	$\frac{\text{rand}()}{\text{RAND_MAX}}$
randomno()

myrand() : "Unif(0,1)".

$X = \text{myrand}()$:

X : Random, $0 \leq X \leq 1$,

$P\{X \leq x\} = x$ if $0 \leq x \leq 1$.

and "independent".

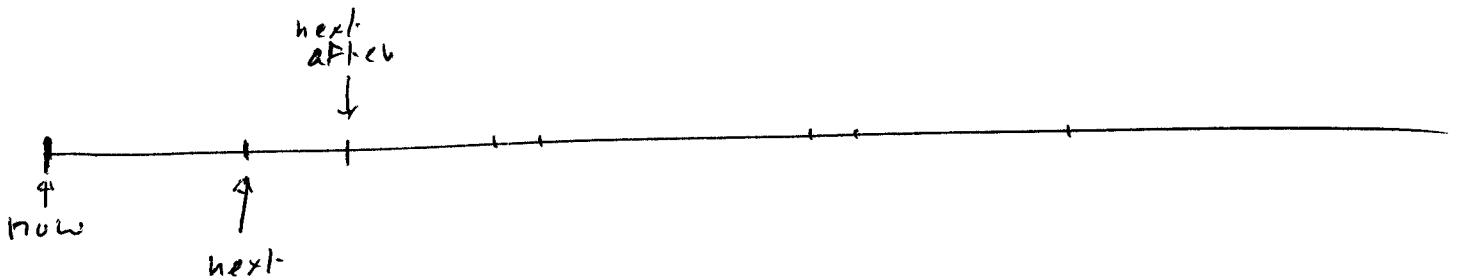
pseudo-random numbers,

(period: 2^{32}).

SimulationData structure : Event list.

(Elem, 1 Object).

Visually :

Events

For example :

- (1) A Frame is flowing from B_1 to L_1 .
At time t_k the Frame will be "all in L_1 " and B_1 must send the next Frame (if any) to L_1 .
- (2) At time t_n a Frame will arrive at N_2 .
- (3) At time t_m there will be a time-out at N_2 .
(check the Frame has been ecked).

Q.
Class: Event.

Many objects of class Event.
(Describe what needs to be done).
(new, delete, constructor, destructor).

During handling an event,
more (new) Events may be
inserted in Event list.

E.g.

A1 make new Frame,
hand to B1.
Set event for time-out.

B1 hand Frame to L1
set "next Frame"
for 1 serialization delay
later.
etc.

Note Book-keeping.

Class PQE

Struct PQE node

Explained Later.

~~Example of Output.~~

~~Constants. h~~ :

Input.b:

includes ~~values~~

values of input variables.

(Linespeed, Packet size, Window,
Loss probability, ...).

Example of Output.