Problem description

- Consider a drive-in restaurant where carhops take orders and bring food to the car.
- Cars arrive in the manner shown in Table 2.11
- There are 2 carhops – Ali and Badu
- Ali is better able to do the job and works a bit faster than Badu.
- The distribution of their service times is shown in Tables 2.12 and 2.13

Bagaimana pembagian beban kerja?

Table 2.11. Interarrival Distribution of Cars

Time between Arrivals (Minutes)	*Probability*	*Cumulative Probability*	*Random-Digit Assignment*
1	0.25	0.25	01–25
2	0.40	0.65	26–65
3	0.20	0.85	66–85
4	0.15	1.00	86–00

Table 2.12. Service Distribution of Ali

Service Time (Minutes)	*Probability*	*Cumulative Probability*	*Random-Digit Assignment*
2	0.30	0.30	01–30
3	0.28	0.58	31–58
4	0.25	0.83	59–83
5	0.17	1.00	84–00

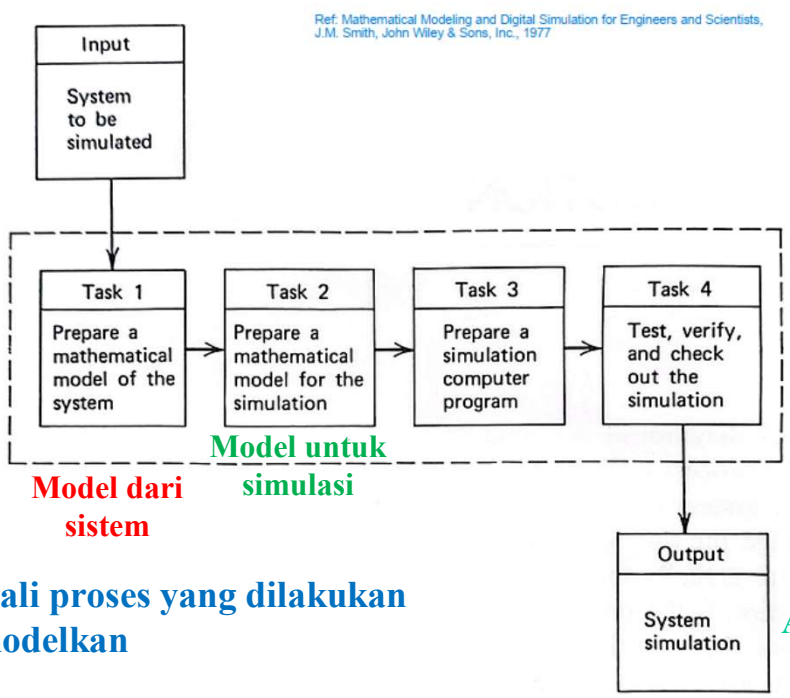
Table 2.13. Service Distribution of Badu

Service Time (Minutes)	*Probability*	*Cumulative Probability*	*Random-Digit Assignment*
3	0.35	0.35	01–35
4	0.25	0.60	36–60
5	0.20	0.80	61–80
6	0.20	1.00	81–00

Problem description

- How well is the current arrangement working in the drive-in restaurant?

Parameter apa yang akan diperlihatkan dari simulasi yang dilakukan untuk menyatakan kinerja dari *drive-in restaurant* tersebut?



Lihat kembali proses yang dilakukan dalam memodelkan

Parameter yang dapat dimunculkan (dari contoh yang lalu untuk 1 server)

- Pengamatan yang dapat diperoleh dari simulasi:
  - **Average waiting time for a customer**
  - Probability that a customer has to wait in the queue
  - **Server idle time (in percentage) → ada Ali dan Badu**
  - Average service time
  - Expected service time
  - **Average time between arrivals**
  - **Average waiting time of those customers who wait**
  - **Average time a customer spends in the queueing system**
  - **……..**

Discrete-event Model

- Model ini sesuai untuk untuk sistem yang perubahan dari keadaan sistemnya (*system state*) hanya terjadi secara diskrit (*discrete points in time*)
- *Basic building block* dari model ini:
  - *Entities and attributes*
  - *Activities and events*
- Fokus pada sistem yang **dinamik** (terkait dengan waktu) dan **stokastik** (ada unsur acaknya)

Discrete-event Model

- **System**
  - A collection of entities
- Model
  - Abstract representation of a system
- System state
  - A collection of variables
- **Entity**
  - Any object or component in the system
- Attributes
- List
- Event
  - An instantaneous occurrence that changes the state of a system
- Event Notice
- Event list
- Activity
- Delay
  - A conditional wait
- Clock
  - A variable representing simulated time

EXAMPLE 3.1 (Ali and Badu , Revised)

Consider the Ali – Badu carhop system of Example 2.2. A discrete-event model has the following components:

System state

$L_Q(t)$, the number of cars waiting to be served at time t

$L_A(t)$, 0 or 1 to indicate Ali being idle or busy at time t

$L_B(t)$, 0 or 1 to indicate Badu being idle or busy at time t

Entities Neither the customers (i.e., cars) nor the servers need to be explicitly represented, except in terms of the state variables, unless certain customer averages are desired

Events

Arrival event

Service completion by Ali

Service completion by Badu

Activities

Interarrival time, defined in Table 2.11

Service time by Ali defined in Table 2.12

Service time by Badu , defined in Table 2.13

Delay A customer's wait in queue until Ali or Badu becomes free

The definition of the model components provides a static description of the model. In addition, a description of the dynamic relationships and interactions between the components is also needed. Some questions that need answers include:

1. How does each event affect system state, entity attributes, and set contents?
2. How are activities defined (i.e., deterministic, probabilistic, or some other mathematical equation)? What event marks the beginning or end of each activity? Can the activity begin regardless of system state, or is its beginning conditioned on the system being in a certain state? (For example, a machining "activity" cannot begin unless the machine is idle, not broken, and not in maintenance.)
3. Which events trigger the beginning (and end) of each type of delay? Under what conditions does a delay begin, or end?
4. What is the system state at time 0? What events should be generated at time 0 to "prime" the model—that is, to get the simulation started?