

# Övning 5, Dator- och telekommunikation, 2014

---

Om vi antar att kunder kommer till ett kösystem med en betjänare och oändligt många buffertplatser så kommer medeltiden i kösystemet att vara

$$E(T) = \frac{1}{\mu} + C \cdot \frac{\lambda}{\mu - \lambda}$$

Här är  $\mu = 1/\text{medelbetjäningstiden}$  och  $C$  är en konstant som inte ändras om man ändrar ankomstintensiteten, men som ändras om de statistiska egenskaperna hos betjäningstiden ändras. En förutsättning är att ankomsterna bildar en poissonprocess. Det innebär något förenklat att om man delar in tidsaxeln i mycket korta intervall av längden  $\Delta$  (så pass korta att  $\Delta \cdot \lambda \ll 1$ ) så ska sannolikheten att en ankomst inträffar i ett intervall vara  $\Delta \cdot \lambda$  oberoende av var intervallet ligger och vad som har hänt tidigare. Detta är ofta en bra approximation.

## Uppgift 1

I ett kösystem enligt ovan så är medeltiden som en kund tillbringar i systemet 2 ms när  $\lambda = 500 \text{ s}^{-1}$ . Medelbetjäningstiden för en kund är 1 ms.

- Beräkna hur lång tid en kund i medeltal väntar i bufferten innan dess betjäning börjar.
- Beräkna medelantal kunder i bufferten.
- Beräkna medeltiden som en kund tillbringar i systemet om  $\lambda$  ökar till  $800 \text{ s}^{-1}$ .

## Uppgift 2

Antag att en utgångsbuffert i en router kan beskrivas med ett kösystem enligt ovan. Det tar i medeltal 0.5 ms att skicka iväg ett IP-paket. När  $\lambda = 1000 \text{ s}^{-1}$  så finns det i medeltal 20 IP-paket i bufferten.

- Beräkna medeltiden som en kund tillbringar i bufferten.
- Beräkna medeltiden som en kund tillbringar i hela kösystemet.
- Beräkna medelantal kunder i hela kösystemet.
- Antag att ankomstintensiteten ökar med 80 %. Hur många kunder finns nu i bufferten i medeltal.

## Uppgift 3

I ett kösystem (= buffert + betjänare) finns det i medeltal 10 kunder. Medelbetjäningstiden är 1 sekund och ankomstintensiteten är  $0.8 \text{ s}^{-1}$ . Inga kunder spärras.

- Beräkna medelantal kunder i betjänares.
- Beräkna medeltiden som en kund tillbringar med att vänta i bufferten.
- Antag att vi begränsar antalet buffertplatser så att det bara får plats en kund i bufferten. Hur påverkar det medelantal kunder som betjänas?

#### Uppgift 4

I ett simuleringsprogram kan händelserna *arrival*, *departure* och *measurement* inträffa. Följande pseudokod visar vad som görs när en av händelserna äger rum.

```
void arrival;
begin
    NumberInSystem := NumberInSystem + 1;
    InsertEvent(departure, time + serviceTime());
    InsertEvent(arrival, time + 4.5);
end;

void departure;
begin
    NumberInSystem := NumberInSystem - 1;
end;

void measurement
begin
    write(utfil, NumberInSystem);
    InsertEvent(measurement, time + 10);
end;
```

Antag att händelselistan se rut så här: (arrival, 5), (measurement, 6), (departure, 9), (departure, 10). När simuleringen startade så var `NumberInSystem = 0`.

- Vilket slags system är det som simuleras?
- Antag att de slumpmässiga betjäningstiderna som ges av metoden `serviceTime()` i tur och ordning är 2, 3, 5, 4, och 1. Hur ser då händelselistan ut efter fyra händelser.
- Antag att medelbetjäningstiden är 4. Hur många kunder kommer då att i medeltal finnas i systemet?

#### Uppgift 5

Du har fått i uppdrag att simulera en ingångsbuffert i en router. Vi antar att det bara finns plats för 20 IP-paket i bufferten och att det alltid tar tiden 1 att betjäna ett IP-paket. Vi är intresserade av att mäta hur många paket det i medeltal finns i bufferten och att bestämma sannolikheten att ett paket spärras för att det inte finns plats.

- Vilka variabler behöver du för att beskriva tillståndet för bufferten?
- Vilka händelser bör finnas i simuleringsprogrammet?
- Skriv pseudokod som visar vad som ska göras vid händelserna i b).

#### Uppgift 6

Ett system består av två ihopkopplade köer enligt nedan. Inga kunder spärras det vill säga buffertarna är oändliga. Vi är intresserade av sannolikheten att det finns mer än 10 kunder i systemet.



- Vilka variabler behövs?
- Vilka händelser?
- Skriv pseudokod för varje händelse.