Scheduling

General purpose system
FIFO
FCFS

RT system
RTEM
↓

I
→ clock-driven – Table Driven
→ event-driven – case switch

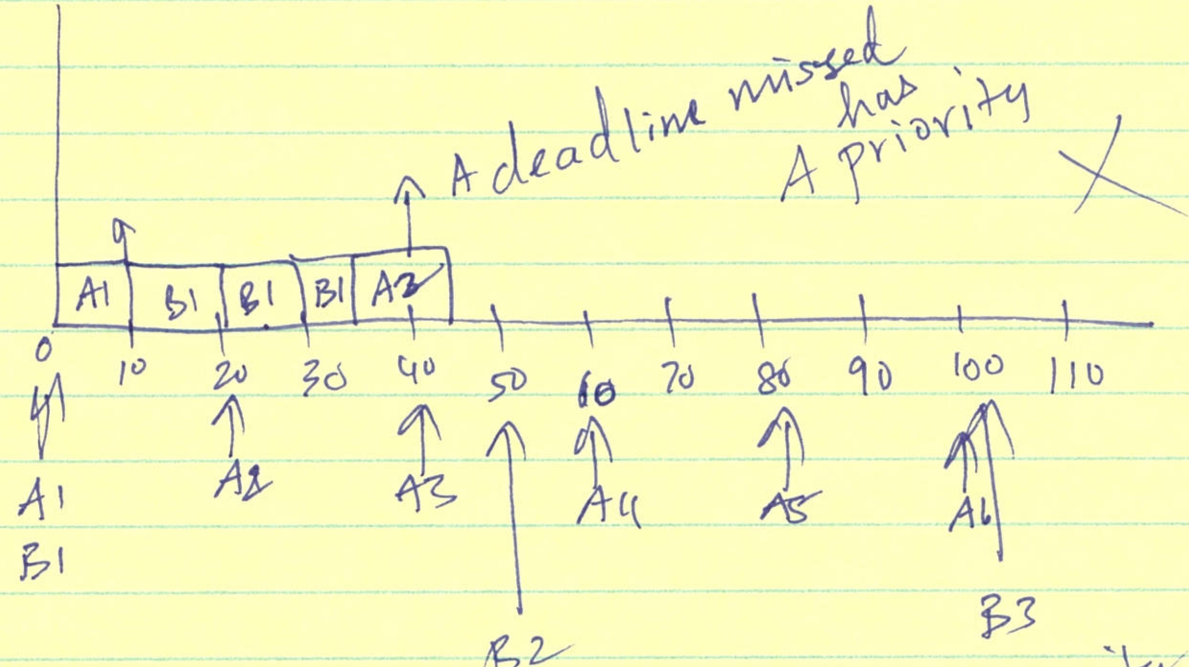
characteristic:
(1) periodic
more than one task
(2) multi-tasking

soln: Deadline driven
(1) – earliest deadline first
EDF

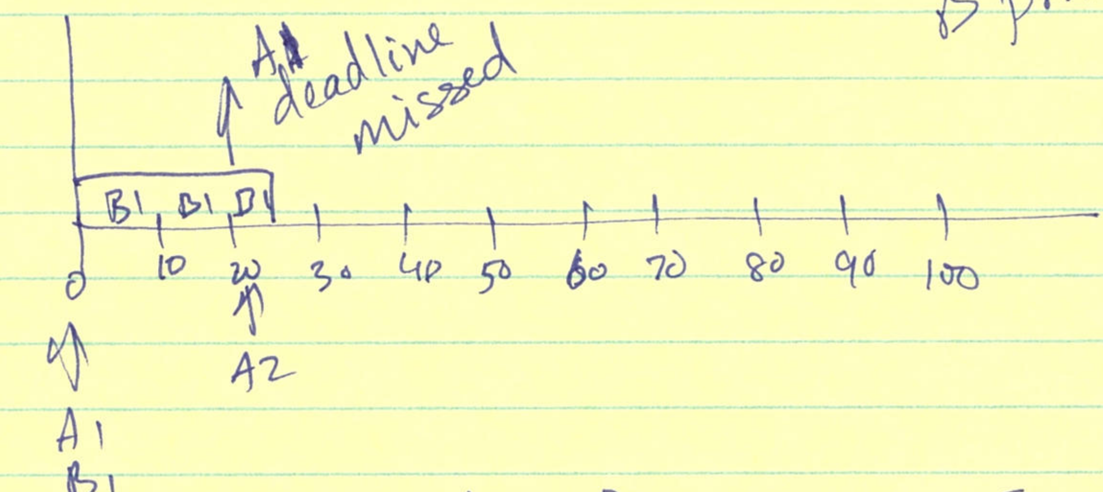
(2) Rate Montonic Scheduling
RMS

II Sharing resources
→ contention for resources
cooperation

III priority – Driven



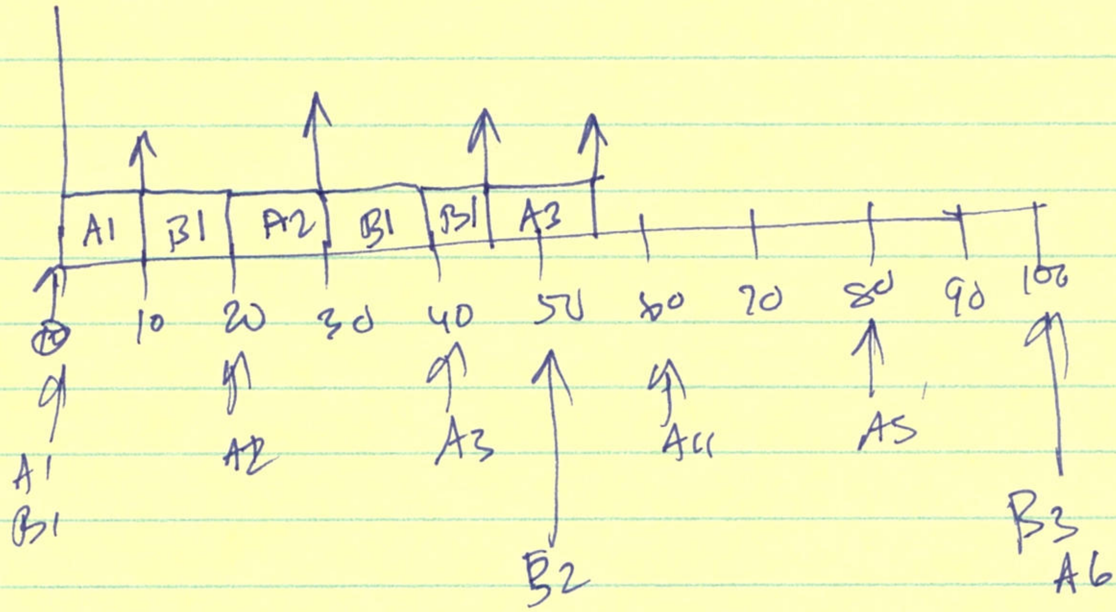
B priority



$$\sum \frac{Ae}{Ap} + \frac{Be}{Bp} = \frac{10}{20} + \frac{25}{50} = 0.5 + 0.5 = 1.0$$

EDF $\sum \frac{ei}{pi} \leq 1.0$

EDF with preemption



Rate montonic scheduling RMS

$$EDF = \sum \frac{e_i}{p_i} \leq 1$$

RMS ① $\sum_{n} \frac{e_i}{p_i} \leq n(2^{1/n}-1)$

where n is the number of tasks

② most frequent task gets the highest priority

higher the frequency $\Rightarrow$ the higher priority

for n = 3,

$n(2^{1/n}-1) = 0.776 \cong 0.78$

Periodic

t_i – task #
r_i – arrival time
e_i = execution time
p_i = period
D_i = relative deadline

3. (20 points) Cyclic Executive and cyclic schedule

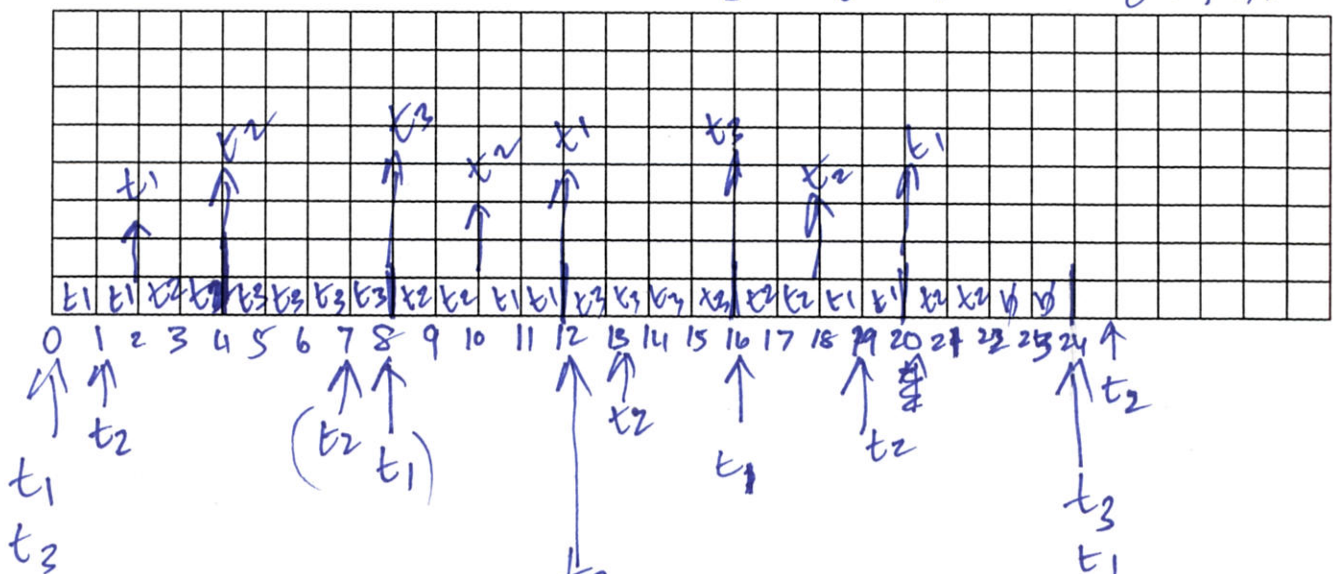
For the task set given below design a cyclic schedule: (i) determine the hyper-period (ii) determine frame size, (iii) provide a timing chart and (iv) a cyclic (executive) schedule.

arrival

ti	ri	ei	pi	Di
t1	0	2	8	6
t2	1	2	6	6
t3	0	4	12	10

(0) $\sum \frac{e_i}{p_i} \le 1$

$= \frac{2}{8} + \frac{2}{6} + \frac{4}{12} = 0.25 + 0.33 + 0.33$
0.91



(i) $H = lcm(8, 6, 12) = 24$

frame size (ii) $f \ge max(e_i) \ge max(2, 2, 4) = (4)$ 5, 6 ...

(iii) verify $2f - gcd(p_i, f) \le D_i$

$8 - gcd(8, 4) \le 6$ $\quad 8 - 4 \le 6$ ✓
$8 - gcd(6, 4) \le 6$ $\quad 8 - 2 \le 6$ ✓
$8 - gcd(12, 4) \le 10$ $\quad 8 - 4 \le 10$ ✓

(iii) See chart.

(iv)

{t1(2); t2(2)};
{t3(4);}
{t2(2); t1(2)};
{t3(4)};
{t2(2); t1(2);}
{t2(2); b(1); b(1)}

1. (25 points) Scheduling: EDS and RMS:

1a. (10 points) Use Earliest Deadline First scheduling policy for the task set given below. Using ***utility*** formula, can it be scheduled satisfactorily using EDS? Prove your answer using a timing chart. Use pre-emptive scheduling; show the schedule two cycles of each task.

1b.(5 points) What is the principle of Rate Monotonic Scheduling (RMS)?

1c. (5 points) Use the RMS analysis to check if the same task set is schedulable with RMS. Why or why not? **No need for the timing chart.**

1d. (5 points) Rewrite the task table given below with updated values such that it is schedulable under RMS.

Task Set		
Task	Period	Execution time
1	3	1
2	4	1
3	10	~~4~~ 2

~~util~~ utility utilization

~~uti~~ $\sum \frac{e_i}{p_i} \leq 1$

EDS it worked

t1	t2	t3	t1	t2	t3	t1	[illegible]	t2	t1	t3													

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

↑ t1 t2 t3 (at 0); ↑ t1 (at 3); ↑ t2 (at 4); ↑ t1 (at 6); ↑ t2 (at 8); ↑ t1 (at 9); ↑ t3 (at 10)

RMS

RMS Rate Monotic Scheduling

① $\sum_n \frac{e_i}{p_i} \leq n(2^{1/n} - 1)$ where n is the number of tasks

$n \to \infty$ 0.69 $n = 3$ 0.776 ≈ 0.78

② most frequent task gets highest priority

$\frac{1}{3} + \frac{1}{4} + \frac{4}{10} = 0.33 + .25 + 0.4 = 0.98$ ≮ 0.776 ≅ 0.78

→ Alt1: $\frac{1}{3} + \frac{1}{4} + \frac{2}{10} = 0.33 + 0.25 + 0.2$ ≤ 0.78 ≤ 0.78

Alt2: $1/3 + 1/4 + \frac{4}{...} = 0.33 + 0.25 + 0.1 = 0.68 < .776$

3a. (15 points) Priority inversion Three real-time tasks bus manager (t1: frequent: high priority), communication task (t2: medium priority) and meteorological task (t3: low priority) run on the NASA Mars Pathfinder. NASA has sent out an urgent request to solve a problem that PathFinder is frequently resetting spontaneously losing all the data. Can you diagnose the problem and solve the problem for NASA?

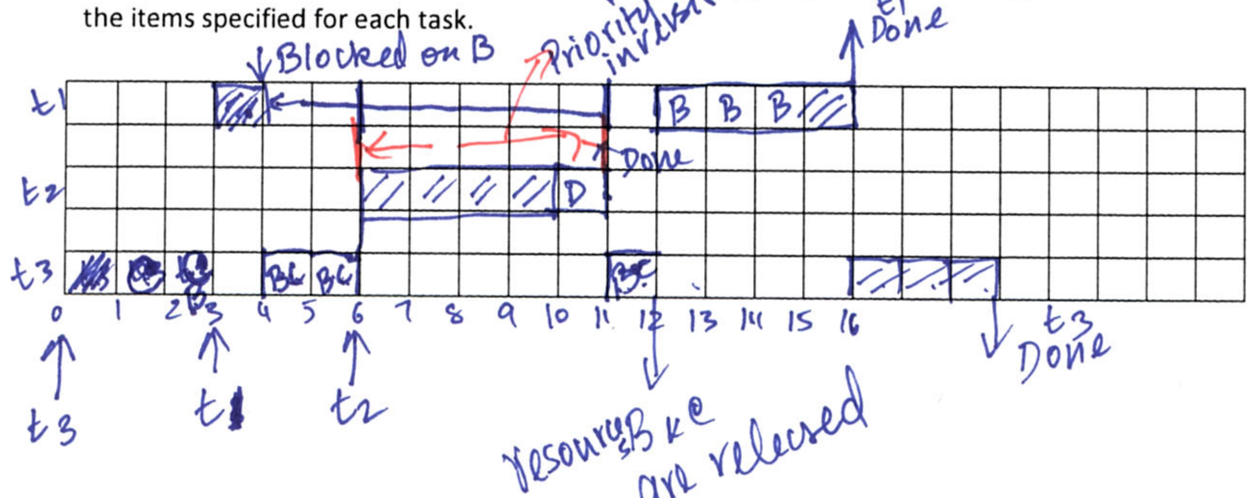
Assume that the tasks with their resource requirements as given below: {arrival time, execution time, priority, resource need}:

(a) t1: { 3, 5, 1:[B:3]} where the task executes for 1 time unit, then requests resource B for 3 time units.

(b) t2: { 6, 5, 2:[D:1]} where the task executes for 4 units and request D for 1 time unit.

(c) t3: { 0, 9, 3:[C:5[B:4]]} where the task executes for 1 unit, then requests resource C, and after 1 time unit requests resource B for 4 times units. (totally 5 time units of resource C)

Construct the schedule for the system using only simple priority policy and **illustrate the occurrence of priority inversion**. {**arrival time, total run time, priority (1 highest), resources**} provide the definition of the items specified for each task.



3b. (10 points) Priority Inheritance

Construct the schedule for the system using priority inheritance and illustrate how it solves the priority inversion problem stated above.