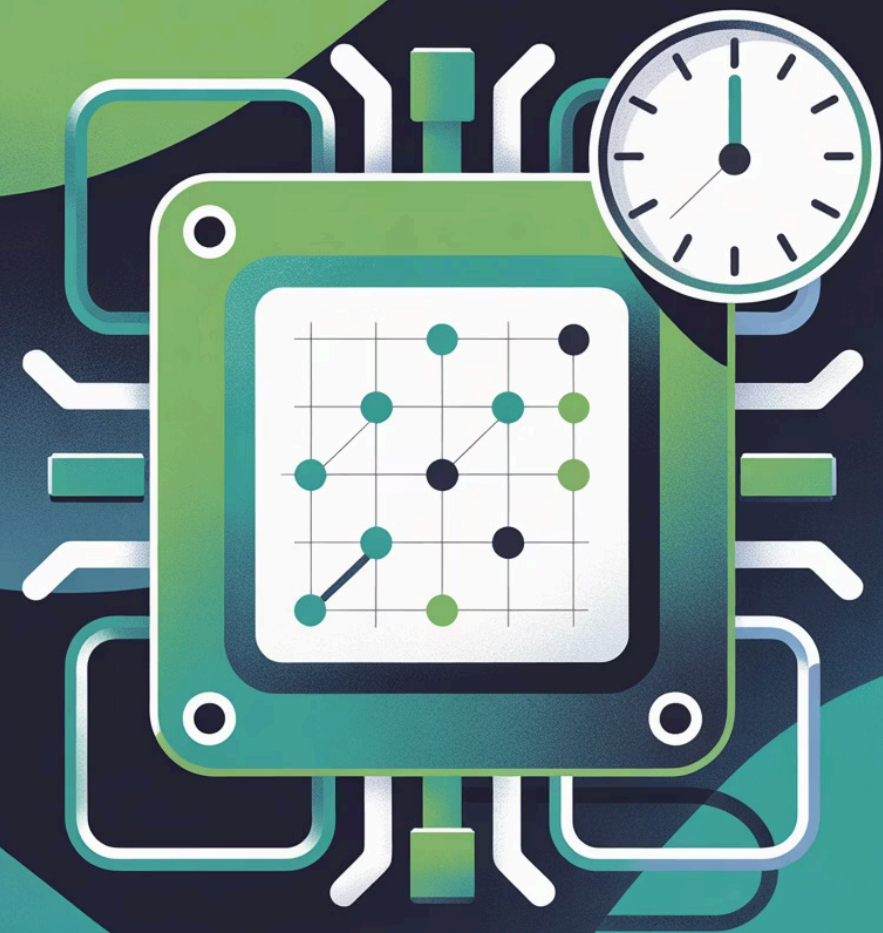


№11 Дәріс. Нақты уақыттағы жоспарлау алгоритмдері (Rate Monotonic, Earliest Deadline First)

Дәріскер: Даркенбаев Д.К. PhD, қауымдастырылған профессор



Дәрістің мақсаты

Нақты уақыт жүйелерінде қолданылатын жоспарлау алгоритмдерінің теориялық негіздерін түсіндіру, Rate Monotonic (RM) және Earliest Deadline First (EDF) алгоритмдерінің жұмыс принциптері, тиімділігі және қолдану аймақтарын меңгерту.

Дәріс жоспары

1. Нақты уақыт жүйелеріне кіріспе
2. Нақты уақыттағы жоспарлау түсінігі
3. Жоспарлау түрлері: статикалық және динамикалық
4. Rate Monotonic (RM) алгоритмі
 - Мәні
 - Формулалары мен шектеулері
 - Мысалдар
5. Earliest Deadline First (EDF) алгоритмі
 - Мәні
 - Артықшылықтары
 - Мысалдар
6. RM және EDF салыстыруы
7. Қолданылу салалары
8. Қорытынды

Кіріспе

Нақты уақыт жүйелерінде (Real-Time Systems) тапсырмалар (tasks) қатаң уақыт шектеулеріне сай орындалуы тиіс.

Мұндай жүйелерде жоспарлау алгоритмдерінің рөлі өте жоғары, себебі олар әр тапсырманың уақытында орындалуын қамтамасыз етеді.

- ❏ **Нақты уақыт жүйелеріндегі басты мақсат** — deadline-ді бұзбай, жүйенің тұрақты жұмысын қамтамасыз ету.



Нақты уақыттағы жоспарлау түрлері

Жоспарлау алгоритмдері екі үлкен санатқа бөлінеді:

Түрі	Сипаттамасы
Статикалық (Static Scheduling)	Тапсырма басымдықтары алдын ала тағайындалады
Динамикалық (Dynamic Scheduling)	Басымдықтар жүйе жұмысы кезінде өзгеруі мүмкін

RM — статикалық; EDF — динамикалық жоспарлау әдісі.

Негізгі түсініктер

Period (T)

Тапсырманың қайталану уақыты

Execution time (C)

Орындауға қажетті уақыт

Deadline (D)

Тапсырма аяқталуы тиіс уақыт

Utilization (U)

Процессор жүктемесі:

$$U = \sum \frac{C_i}{T_i}$$

Rate Monotonic Scheduling (RM)

Мәні

Rate Monotonic (RM) — ең қысқа периодты тапсырмаға ең жоғары басымдық беретін статикалық жоспарлау алгоритмі.

1

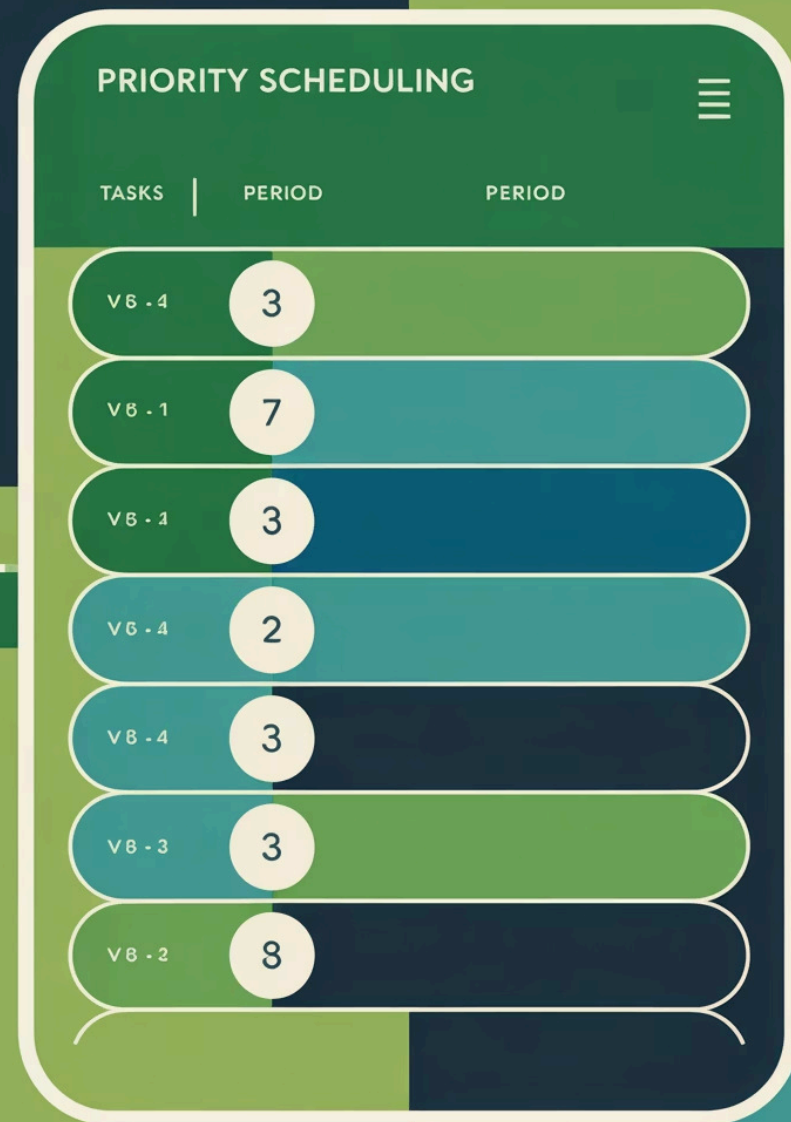
Т аз (жиі орындалады)

Жоғары приоритет

2

Т көп

Төмен приоритет



RM қолдану шарты

RM алгоритмі тек периодтық тапсырмаларға арналған.

Жүктемені тексеру формуласы (процессор утилизациясы):

$$U \leq n(2^{1/n} - 1)$$

Мысалы:

Тапсырма саны (n)	Максималды U
1	1.00
2	0.828
3	0.78
→ шексіздік	0.693

❏ Яғни, 69.3% жүктемені қамтамасыз етеді.

RM Мысалы

Екі тапсырма:

Тапсырма	C	T	C/T
τ1	1	4	0.25
τ2	1	5	0.2

$$U = 0.25 + 0.2 = 0.45$$

Екі тапсырма үшін шекті мән 0.828, сондықтан жүйе қанағаттандырады.

Earliest Deadline First (EDF)

Мәні

Earliest Deadline First — ең жақын Deadline-ге ие тапсырмаға басымдық беретін динамикалық алгоритм.



Тез бітетін

Жоғары басымдық



Уақыт көп

Төмен басымдық

EDF Артықшылығы және Мысалы

Артықшылығы

CPU жүктемесі 100% дейін жетеді

$$U \leq 1.0$$

Мысал

Алдыңғы екі тапсырманы алайық:

$$U = 0.45 \leq 1.0$$

EDF — гарантты орындайды (RM-ге қарағанда тиімдірек).

RM vs EDF салыстыру

Критерий	RM	EDF
Басқару	Статикалық	Динамикалық
Тиімділік	69.3%	100%
Жүзеге асыру	Оңай	Күрделі
Priority берілуі	Периодқа байланысты	Deadline-ге байланысты
Қолданылуы	Safety-critical systems	Flexible real-time systems

Қолданылу аясы

Жүйе түрі	Алгоритм	Мысалы
Қатаң RT (hard real-time)	RM	Авиация, ядролық жүйелер
Жұмсақ RT (soft real-time)	EDF	Медиа, робототехника, IoT

Қорытынды

Нақты уақыт жүйелері үшін жоспарлау алгоритмдерінің дұрыстығы жүйенің сенімді және қателіксіз жұмыс істеуін қамтамасыз етеді.

Rate Monotonic (RM)

Статикалық басымдық беру әдісі, қарапайым және сенімді, бірақ процессорды толық пайдалану деңгейі шектеулі (~69%). Қатаң нақты уақыттық жүйелерде кең қолданылады.

Earliest Deadline First (EDF)

Динамикалық жоспарлау әдісі, процессорды 100% жүктеуге мүмкіндік береді, бірақ іске асыруы күрделірек. Жұмсақ нақты уақыттық жүйелерде жиі қолданылады.

Екі алгоритм де нақты уақыт жүйелерінің уақтылық талаптарын қамтамасыз етуге арналған. Жүйе талаптарына байланысты сәйкес әдіс таңдалады.

Қысқаша тұжырым

- RM — статикалық, периодқа сүйенеді
- EDF — динамикалық, ең ерте deadline-ді таңдайды
- EDF тиімді, бірақ RM сенімді және қарапайым
- Процессор жүктемесі $RM: \leq 69.3\%$, $EDF: \leq 100\%$

Бақылау сұрақтары

1. Нақты уақыттағы жүйелердің негізгі ерекшеліктері қандай?
2. RM және EDF алгоритмдерінің айырмашылығы неде?
3. Неліктен RM статикалық алгоритм деп аталады?
4. RM-де процессор жүктемесін тексеру формуласын жазыңыз.
5. Қай жағдайда EDF қолдану тиімді?



Пайдаланылған әдебиеттер

1. Liu C. L., Layland J. W. "Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment." Journal of the ACM, 1973.
2. Rajkumar R. Real-Time Systems. McGraw-Hill, 1999.
3. Jane W. S. Liu. Real-Time Systems. Prentice Hall, 2000.
4. Buttazzo G. Hard Real-Time Computing Systems. Springer, 2011.
5. Таненбаум Э., Бос Х. Современные операционные системы. — Москва: Вильямс, 2021.
6. K. Kim, Embedded and Real-Time Systems, Springer, 2019.
7. Lecture notes from MIT OpenCourseWare – Real-Time Systems.