Параллельные вычислительные процессы

1

РОМАНЕНКО ВЛАДИМИР ВАСИЛЬЕВИЧ,
К.Т.Н., ДОЦЕНТ КАФ. АСУ ТУСУР

Параллельные вычислительные процессы

ОПИСАНИЕ ВЫЧИСЛИТЕЛЬНЫХ ПРОЦЕССОВ

Базовые определения

- Имена процессов будем обозначать словами, составленными из прописных букв, а буквами P, Q, R, … будем обозначать произвольные процессы.
- Буквы x, y, z, … используются для переменных, обозначающих события.
- Буквы A, B, C, … используются для обозначения множества событий.
- Буквы X, Y, Z, … используются для переменных, обозначающих процессы.
- Алфавит процесса P обозначается αP.
- Процесс с алфавитом αP, такой, что в нем не происходит ни одно событие из αP, назовем $СТОП_{\alpha P}$.

Базовые определения

Пример: автомат, торгующий шоколадками.

- В качестве имени процесса выберем *ТАП* (торговый аппарат простой).
- Имена событий – *мон* (опускание монеты в щель автомата) и *шок* (появление шоколадки из выдающего устройства).
- Алфавит α*ТАП* = {*мон*, *шок*}.

Префиксная запись

Префиксная форма описания процессов:

$$(x \to P),$$

где

- x – событие;
- P – процесс.

При этом

$$\alpha(x \to P) = \alpha P,\ x \in \alpha P.$$

Пример:

$$(\text{мон} \to (\text{шок} \to (\text{мон} \to (\text{шок} \to \text{СТОП}_{\alpha ТАП})))).$$

Рекурсивная запись

Рекурсивный метод определения процесса:

$$P = (x \rightarrow P),$$

$$P = (x \rightarrow (y \rightarrow P)),$$

и т.д. Здесь x, $y \in \alpha P$.

Пример 1:

$$\mathit{ТАП} = (\mathit{мон} \rightarrow (\mathit{шок} \rightarrow \mathit{ТАП})).$$

Рекурсивная запись

Рекурсивный метод определения процесса:

$$P = (x \to P),$$

$$P = (x \to (y \to P)),$$

и т.д. Здесь x, $y \in \alpha P$.

Пример 2: процесс *ЧАСЫ* описывает часы, единственная функция которых – тикать:

$$\alpha ЧАСЫ = \{тик\},$$

$$ЧАСЫ = (тик \to ЧАСЫ).$$

Определение выбора

Описание объектов с несколькими линиями поведения:

$$(x \to P \mid y \to Q),$$

где

$$\alpha(x \to P \mid y \to Q) = \alpha P,\ x, y \in \alpha P,\ \alpha P = \alpha Q.$$

Пример 2: копирование битов из входного канала в выходной:

$$\alpha КОПИБИТ = \{вв.0,\ вв.1,\ выв.0,\ выв.1\},$$

$$КОПИБИТ = ((вв.0 \to выв.0 \mid вв.1 \to выв.1) \to КОПИБИТ).$$

Параллельные процессы

Оператор параллельной композиции:

$$P \mid\mid Q.$$

Законы:

- $P \mid\mid Q = Q \mid\mid P$ – логическая симметрия между процессом и его окружением;
- $(c \to P) \mid\mid (c \to Q) = c \to (P \mid\mid Q)$,
$(c \to P) \mid\mid (d \to Q) = СТОП$ – пара процессов с одинаковыми алфавитами либо одновременно выполняет одно и то же действие, либо попадает в состояние тупика, если начальные события процессов не совпадают;

Параллельные процессы

Оператор параллельной композиции:

$$P \,||\, Q.$$

Законы:

- $P \,||\, (Q \,||\, R) = (P \,||\, Q) \,||\, R$ – ассоциативность (при совместной работе процессов неважно, в каком порядке они объединены оператором параллельной композиции);
- $P \,||\, СТОП_{\alpha P} = СТОП_{\alpha P}$ – процесс, находящийся в тупиковой ситуации, приводит к тупику всей системы.

Задача об обедающих философах

Алфавит философа:

$\alpha ФИЛ_i = \{i.садится,\ i.встает,\ i.берет\_вил.i,\ i.берет\_вил.(i+_5 1),\ i.кладет\_вил.i,\ i.кладет\_вил.(i+_5 1)\}$

Алфавит вилки:

$\alpha ВИЛ_i = \{i.берет\_вил.i,\ (i-_5 1).берет\_вил.i,\ i.кладет\_вил.i,\ (i-_5 1).кладет\_вил.i\}$

Задача об обедающих философах

Поведение философа:

$$ФИЛ_i = (i.садится \to i.берет\_вил.i \to i.берет\_вил.(i+_5 1) \to i.кладет\_вил.i \to i.кладет\_вил.(i+_5 1) \to i.встает \to ФИЛ_i)$$

Поведение вилки:

$$ВИЛ_i = (i.берет\_вил.i \to i.кладет\_вил.i \to ВИЛ_i \mid (i-_5 1).берет\_вил.i \to (i-_5 1).кладет\_вил.i \to ВИЛ_i)$$

Задача об обедающих философах

Поведение всего пансиона:

$$\text{\textit{ФИЛОСОФЫ}} = (\text{\textit{ФИЛ}}_0 \,||\, \text{\textit{ФИЛ}}_1 \,||\, \text{\textit{ФИЛ}}_2 \,||\, \text{\textit{ФИЛ}}_3 \,||\, \text{\textit{ФИЛ}}_4),$$

$$\text{\textit{ВИЛКИ}} = (\text{\textit{ВИЛ}}_0 \,||\, \text{\textit{ВИЛ}}_1 \,||\, \text{\textit{ВИЛ}}_2 \,||\, \text{\textit{ВИЛ}}_3 \,||\, \text{\textit{ВИЛ}}_4),$$

$$\text{\textit{ПАНСИОН}} = (\text{\textit{ФИЛОСОФЫ}} \,||\, \text{\textit{ВИЛКИ}})$$

Протоколы поведения процессов

Протоколом поведения процесса *называется конечная последовательность символов, фиксирующая события, в которых процесс участвовал до некоторого момента времени.*

Примеры:

- $\langle\rangle$ – пустой протокол;
- $\langle x \rangle$ – протокол из одного события;
- $\langle x, y \rangle$ – протокол из двух событий и т.д.
- $\langle \textit{мон, шок, мон} \rangle$, $\langle \textit{мон, шок, мон, шок} \rangle$.

Операции с протоколами

Конкатенация:

$$s\hat{}t,$$

$$t^{n+1} = t\hat{}t^{n},$$

$$(s\hat{}t)^{n+1} = s\hat{}(s\hat{}t)^{n}\hat{}t.$$

Например:

$$\langle \text{мон}, \text{шок} \rangle \hat{} \langle \text{мон} \rangle = \langle \text{мон}, \text{шок}, \text{мон} \rangle .$$

Свойства:

- $s\hat{}(t\hat{}u) = (s\hat{}t)\hat{}u$ – ассоциативность;
- $s\hat{}\langle\rangle = \langle\rangle\hat{}s = s$ – пустой протокол служит единицей.

Операции с протоколами

Сужение:

$$t \uparrow A.$$

Например:

$$\langle мон, шок, мон \rangle \uparrow \langle мон \rangle = \langle мон, мон \rangle.$$

Свойства:

- $\langle x \rangle \uparrow A = \langle x \rangle$, если $x \in A$, $\langle y \rangle \uparrow A = \langle \rangle$, если $y \notin A$;
- $(s \hat{\ } t) \uparrow A = (s \uparrow A) \hat{\ } (t \uparrow A)$ – дистрибутивность;
- $\langle \rangle \uparrow A = \langle \rangle$, $s \uparrow \varnothing = \langle \rangle$, $(s \uparrow A) \uparrow B = s \uparrow (A \cap B)$.

Операции с протоколами

Голова и хвост:

s_{o} – первый элемент;

s' – результат, полученный после его удаления,

т.е. $\langle x, y\rangle_{\text{o}} = x$, $\langle x, y\rangle' = \langle y\rangle$.

Например:

$\langle \textit{мон}, \textit{шок}, \textit{мон}\rangle_{\text{o}} = \textit{мон}$,

$\langle \textit{мон}, \textit{шок}, \textit{мон}\rangle' = \langle \textit{шок}, \textit{мон}\rangle$.

Параллельные вычислительные процессы

СЕТИ ПЕТРИ

Основные определения

Сеть Петри N является четверкой $N = (P, T, I, O)$, где:

- $P = \{p_1, p_2, \ldots, p_n\}$ – конечное множество *позиций*, $n \geq 0$;
- $T = \{t_1, t_2, \ldots, t_m\}$ – конечное множество *переходов*, $m \geq 0$;
- $I: T \rightarrow P^*$ – входная функция, сопоставляющая переходу мультимножество его входных позиций;
- $O: T \rightarrow P^*$ – выходная функция, сопоставляющая переходу мультимножество его выходных позиций.

Основные определения

Граф сети Петри обладает двумя типами узлов: *кружок*, представляющий позицию сети Петри; и *планка*, представляющая переход сети Петри.

Маркировка μ – функция отображения

$$\mu: P \to Nat,$$

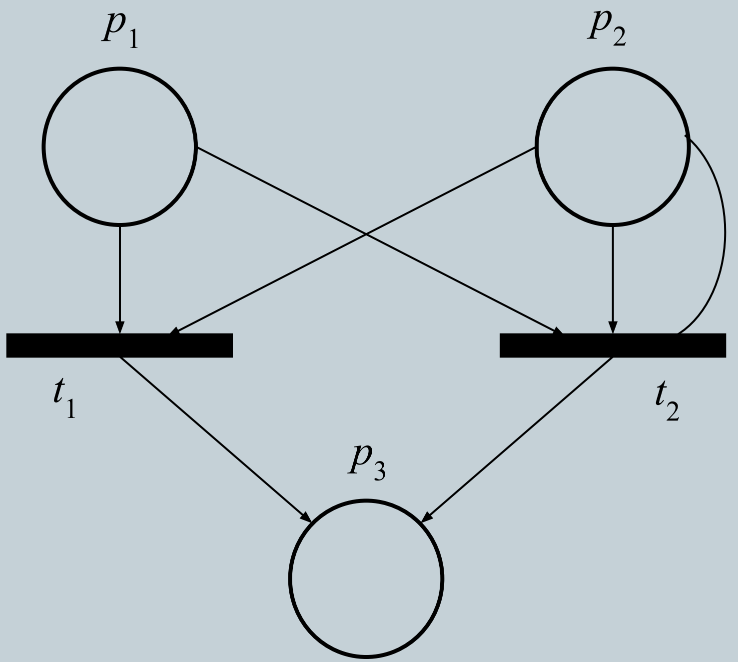
$$\mu = \langle \mu(p_1), \mu(p_2), \ldots, \mu(p_n) \rangle,$$

где n – число позиций в сети Петри и $\mu(p_i) \in Nat$, $1 \le i \le n$ – количество фишек в позиции p_i.

Основные определения

Пример: Сеть Петри $N = (P, T, I, O)$,

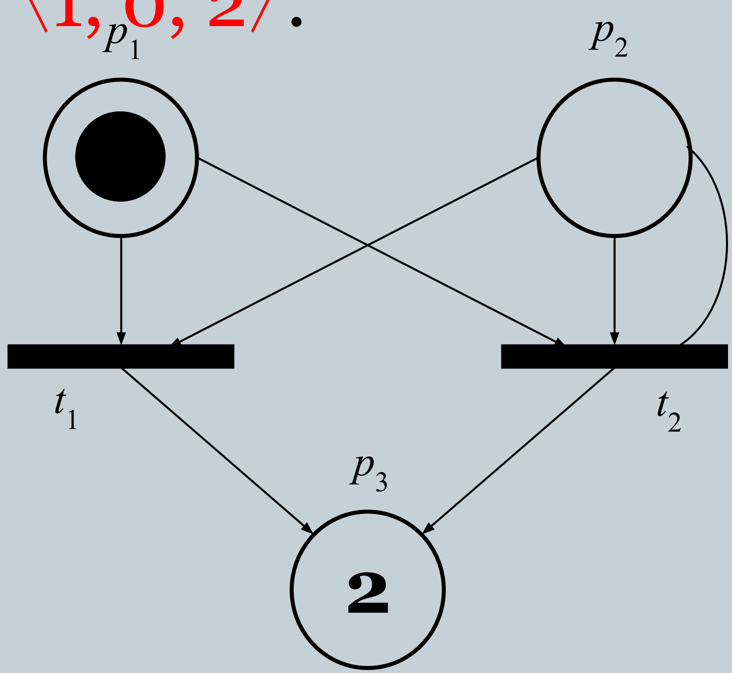
- $P = \{p_1, p_2, p_3\}$,
- $T = \{t_1, t_2\}$,
- $I(t_1) = \{p_1, p_1, p_2\}$, $O(t_1) = \{p_3\}$,
- $I(t_2) = \{p_1, p_2, p_2\}$, $O(t_2) = \{p_3\}$.



Основные определения

Маркированная сеть Петри $N = (P, T, I, O, \mu)$ определяется совокупностью структуры сети Петри $N = (P, T, I, O)$ и маркировки μ.

Например, $\mu = \langle 1, 0, 2 \rangle$:



Основные определения

Матричный вид сети Петри $N = (P, T, I, O)$ задается парой (D^{-}, D^{+}), где:

- $D^{-}[k, i] = \hat{}\#(p_i, t_k)$ – кратность дуги, ведущей из позиции p_i в переход t_k;
- $D^{+}[k, i] = \#\hat{}(t_k, p_i)$ – кратность дуги, ведущей из перехода t_k в позицию p_i,

для произвольных $1 \le k \le m$, $1 \le i \le n$. При этом

$$\mu' = \mu - e[k]D^{-} + e[k]D^{+} = \mu + e[k]D.$$

Запуск сетей Петри

Сеть Петри *выполняется* посредством *запусков* переходов. При этом образуется новая маркировка μ':

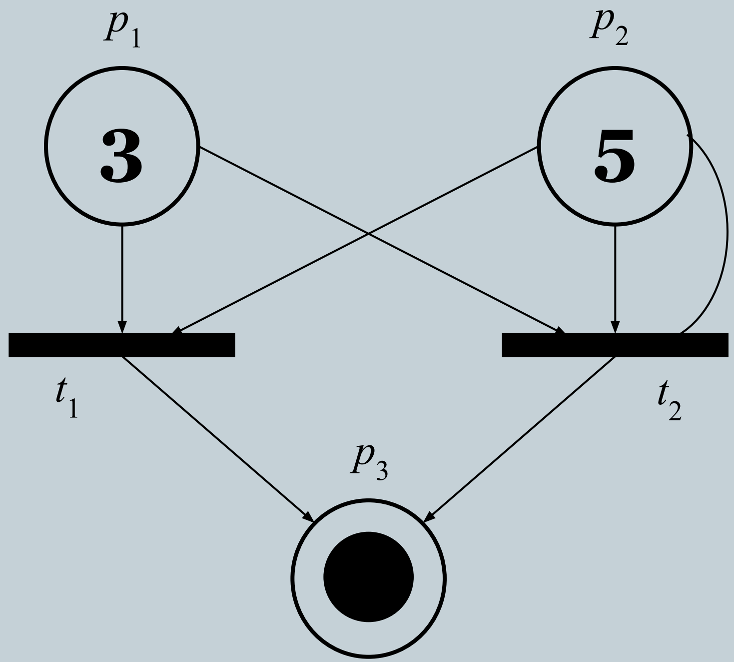
$$\mu'(p) = \mu(p) - \hat{}\#(p, t) + \#\hat{}(t, p),$$

где:

- $\hat{}\#: P \times T \to Nat$;
- $\#\hat{}: T \times P \to Nat$;
- $\mu(p) \geq \hat{}\#(p, t)$;
- $p \in P$.

Запуск сетей Петри

Пример: $\mu = \langle 3, 5, 1 \rangle$

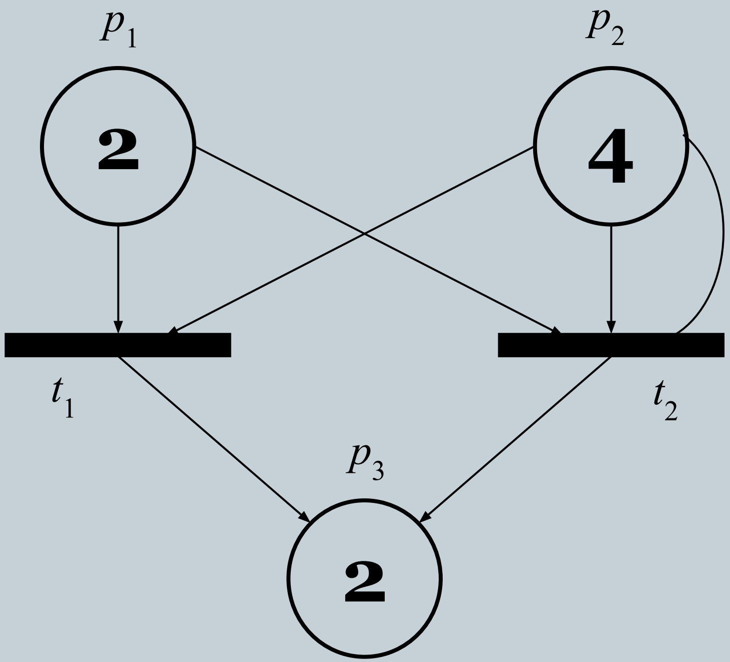


Запуск сетей Петри

Пример: $\mu = \langle 3, 5, 1 \rangle$

После перехода t_1:

$\mu' = \langle 2, 4, 2 \rangle$



Запуск сетей Петри

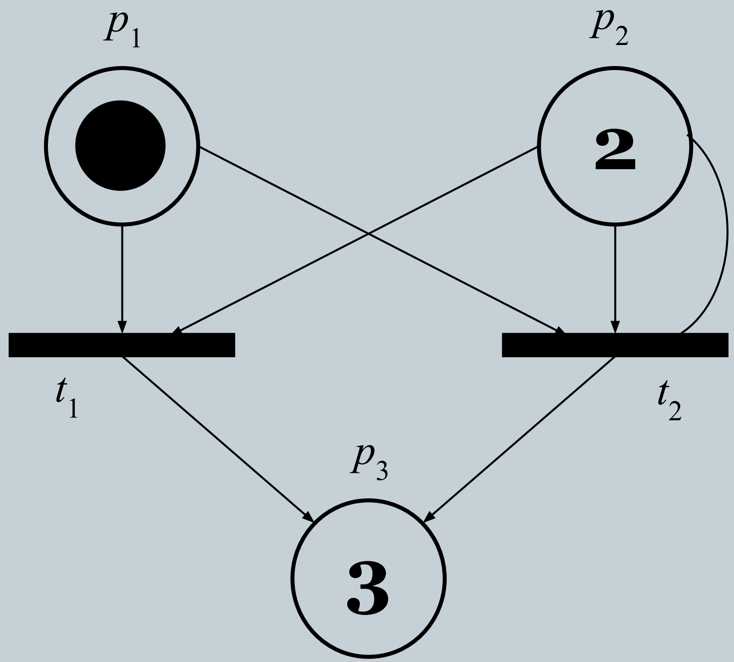
Пример: $\mu = \langle 3, 5, 1 \rangle$

После перехода t_1:

$\mu' = \langle 2, 4, 2 \rangle$

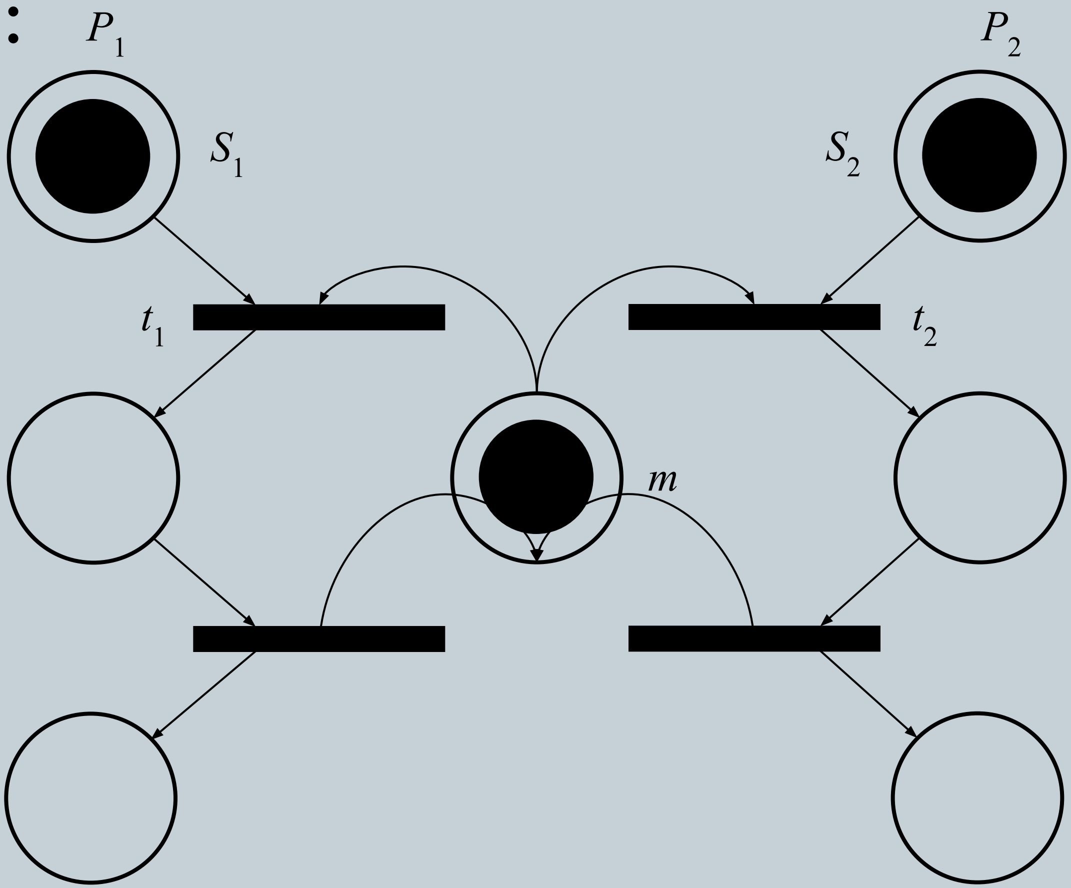
После перехода t_2:

$\mu'' = \langle 1, 2, 3 \rangle$



Моделирование систем

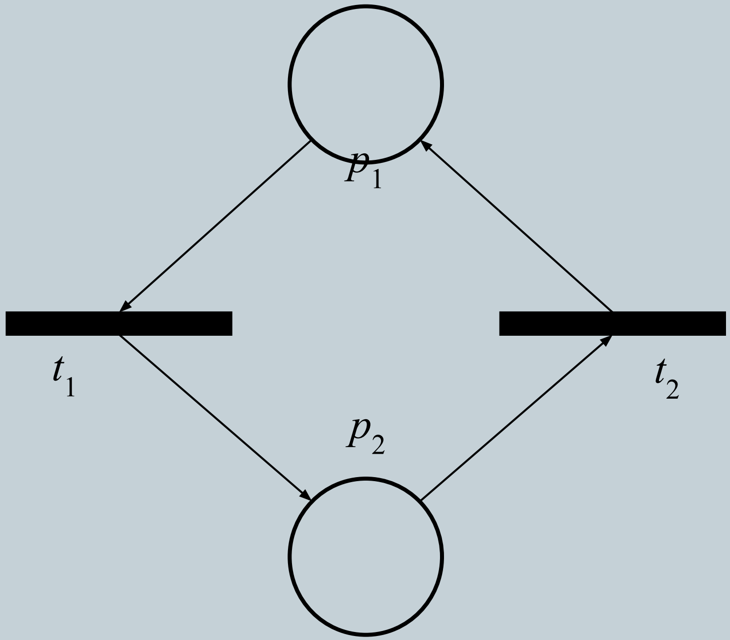
Механизм взаимного исключения для двух процессов:



Простой семафор S:

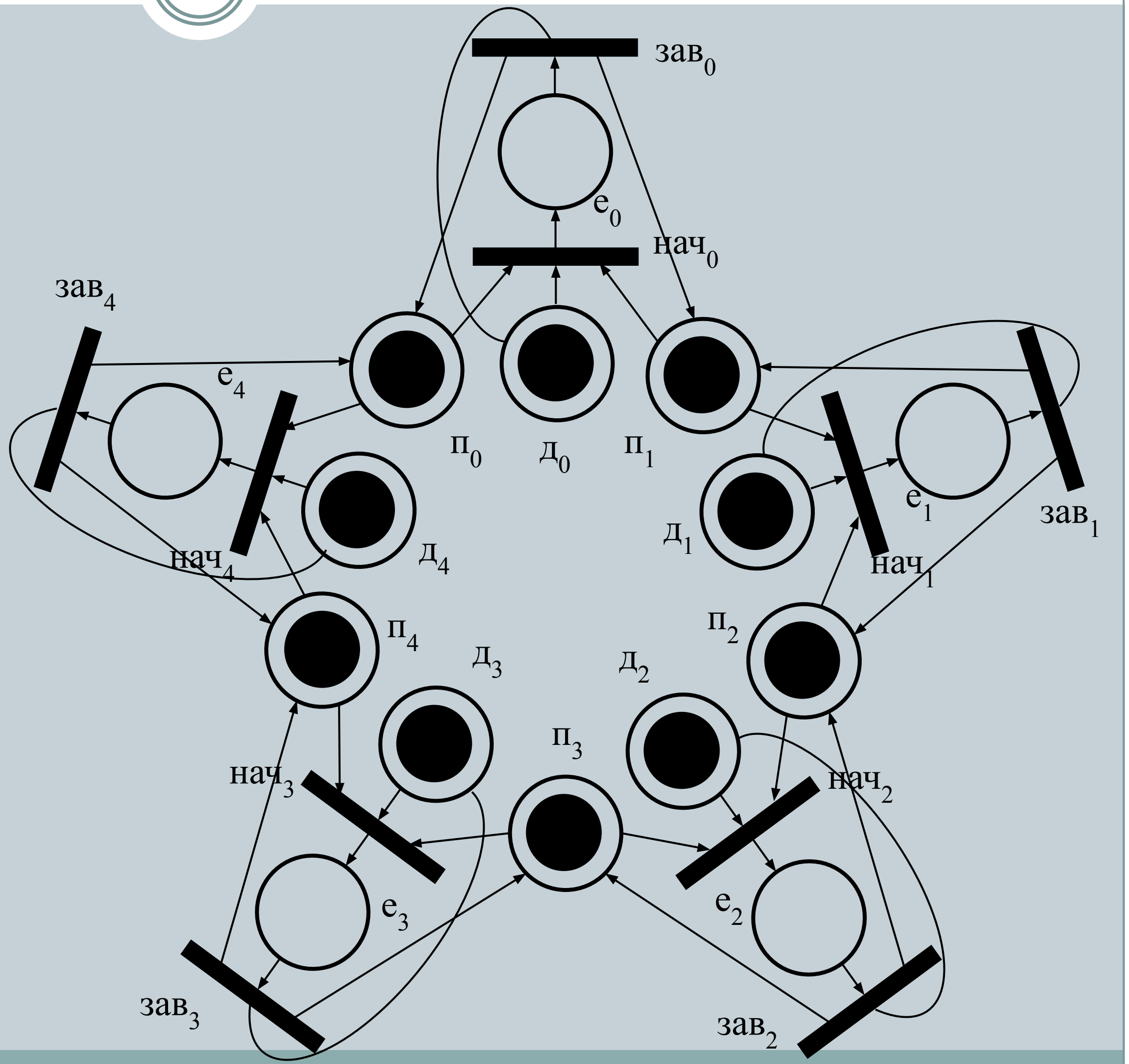
- n – текущее значение счётчика;
- N_{max} – максимальное значение счётчика;
- P – операция блокирования семафора (WaitOne, WaitForSingleObject, acquire, …);
- V – операция деблокирования семафора (Release, ReleaseSemaphore, release, …).

Простой семафор S:



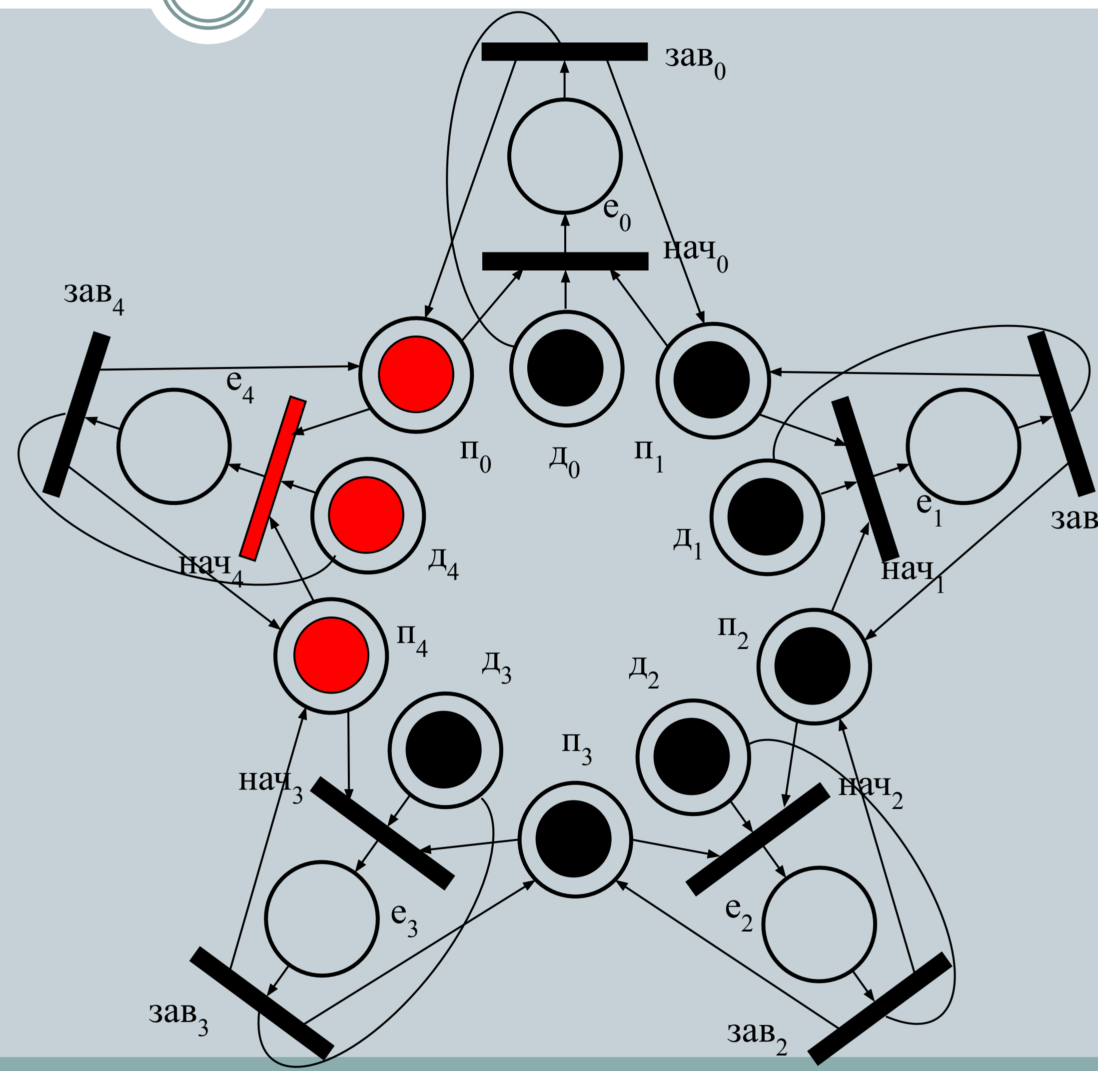
Моделирование систем

Задача об обедающих философах:



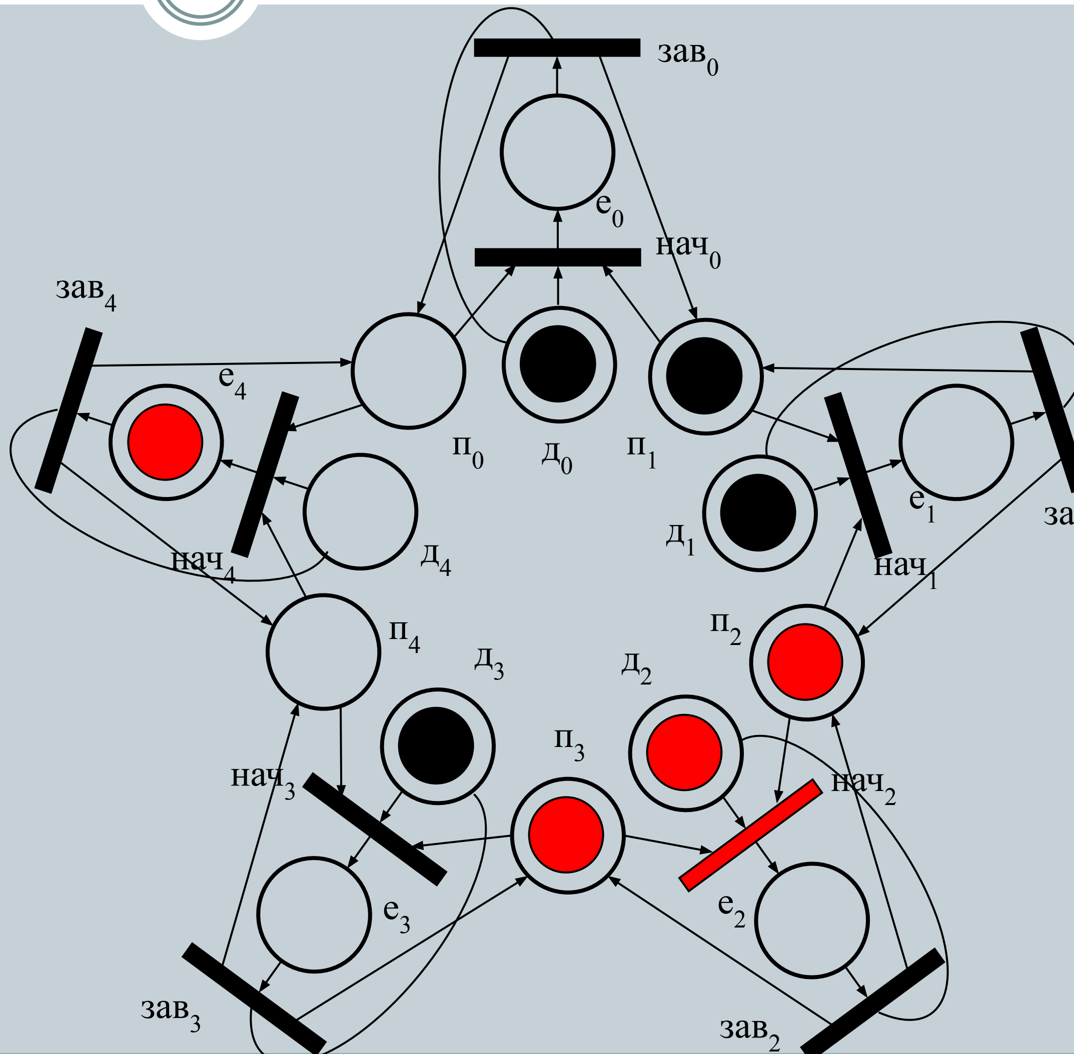
Моделирование систем

Задача об обедающих философах:



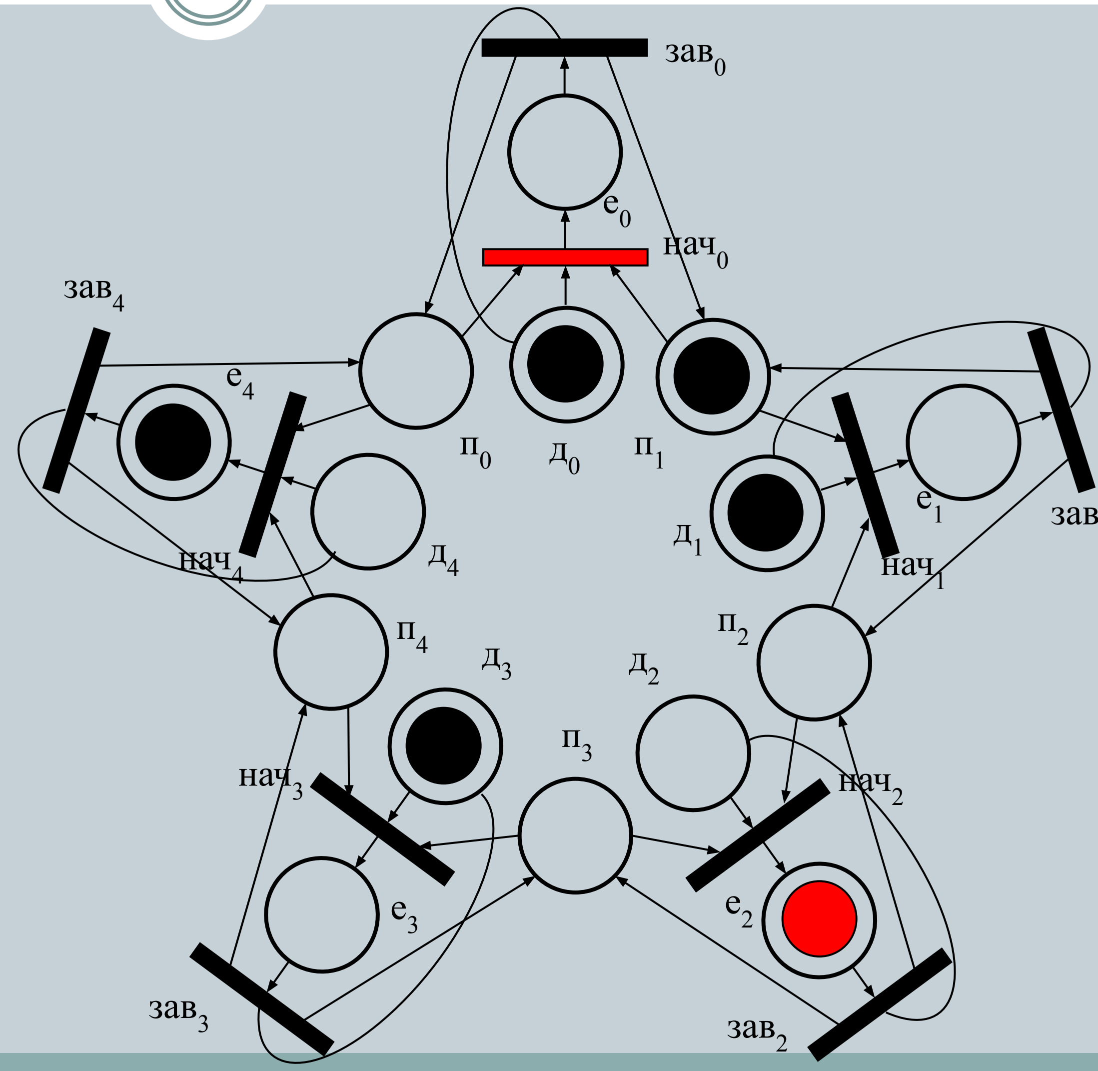
Моделирование систем

Задача об обедающих философах:



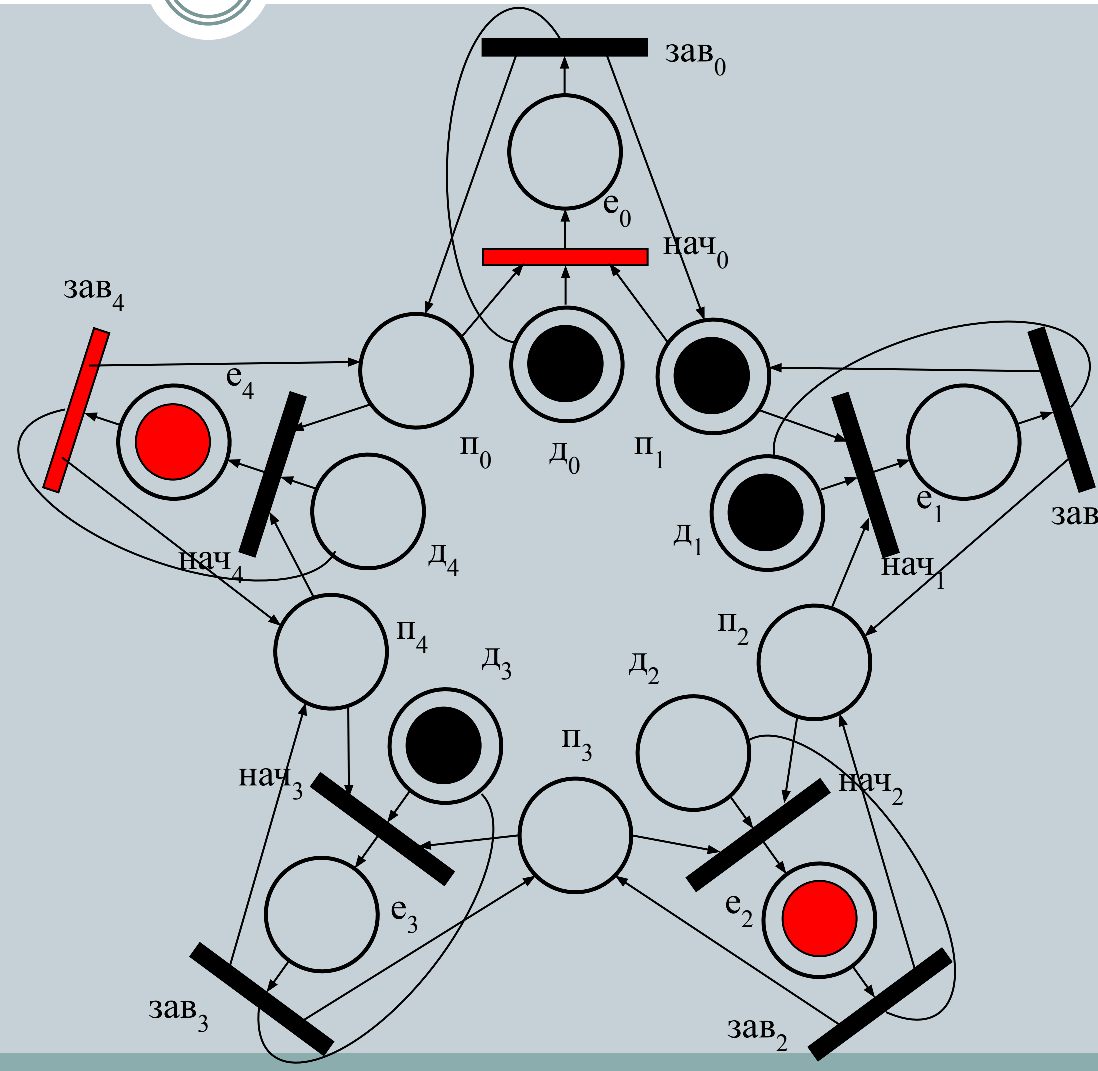
Моделирование систем

Задача об обедающих философах:



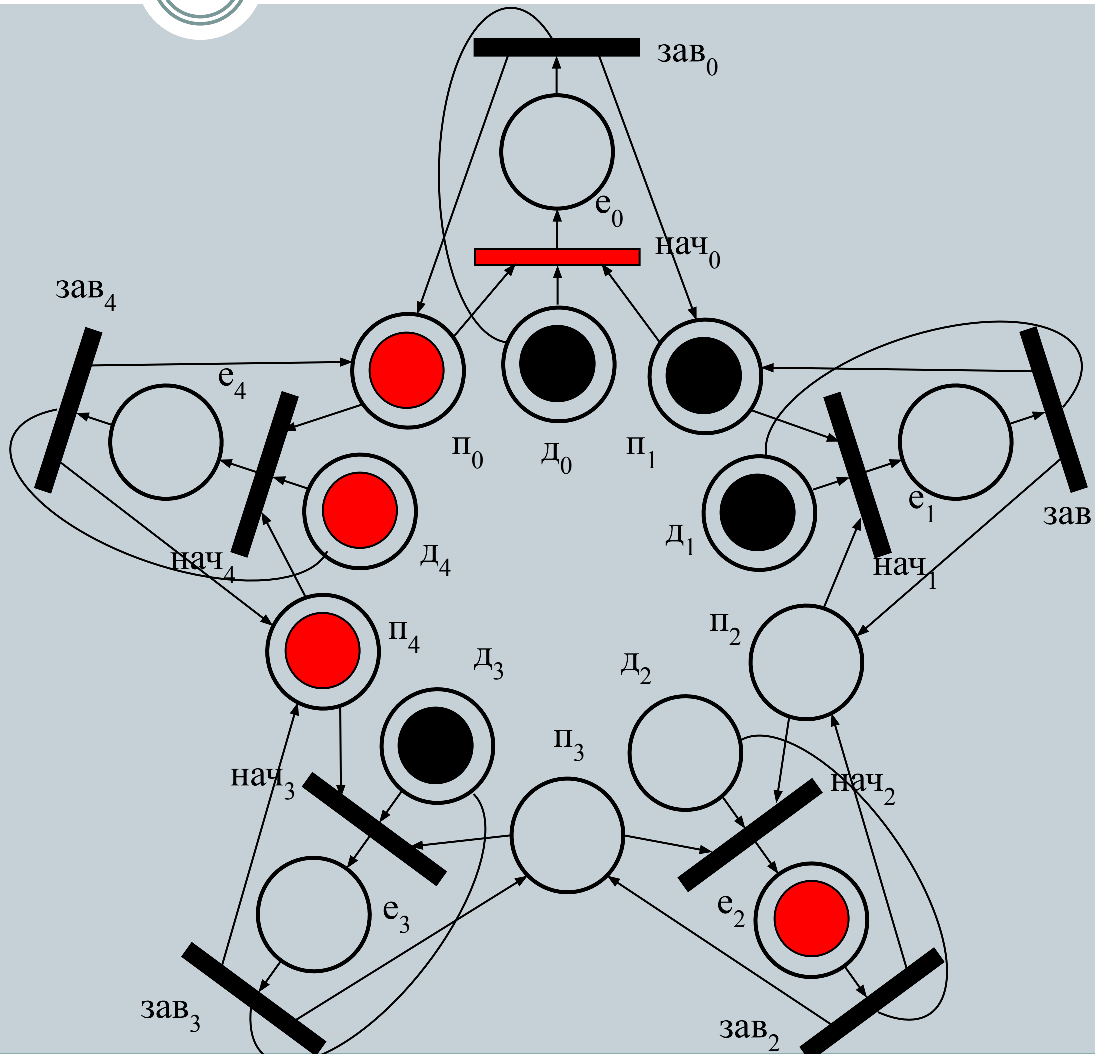
Моделирование систем

Задача об обедающих философах:



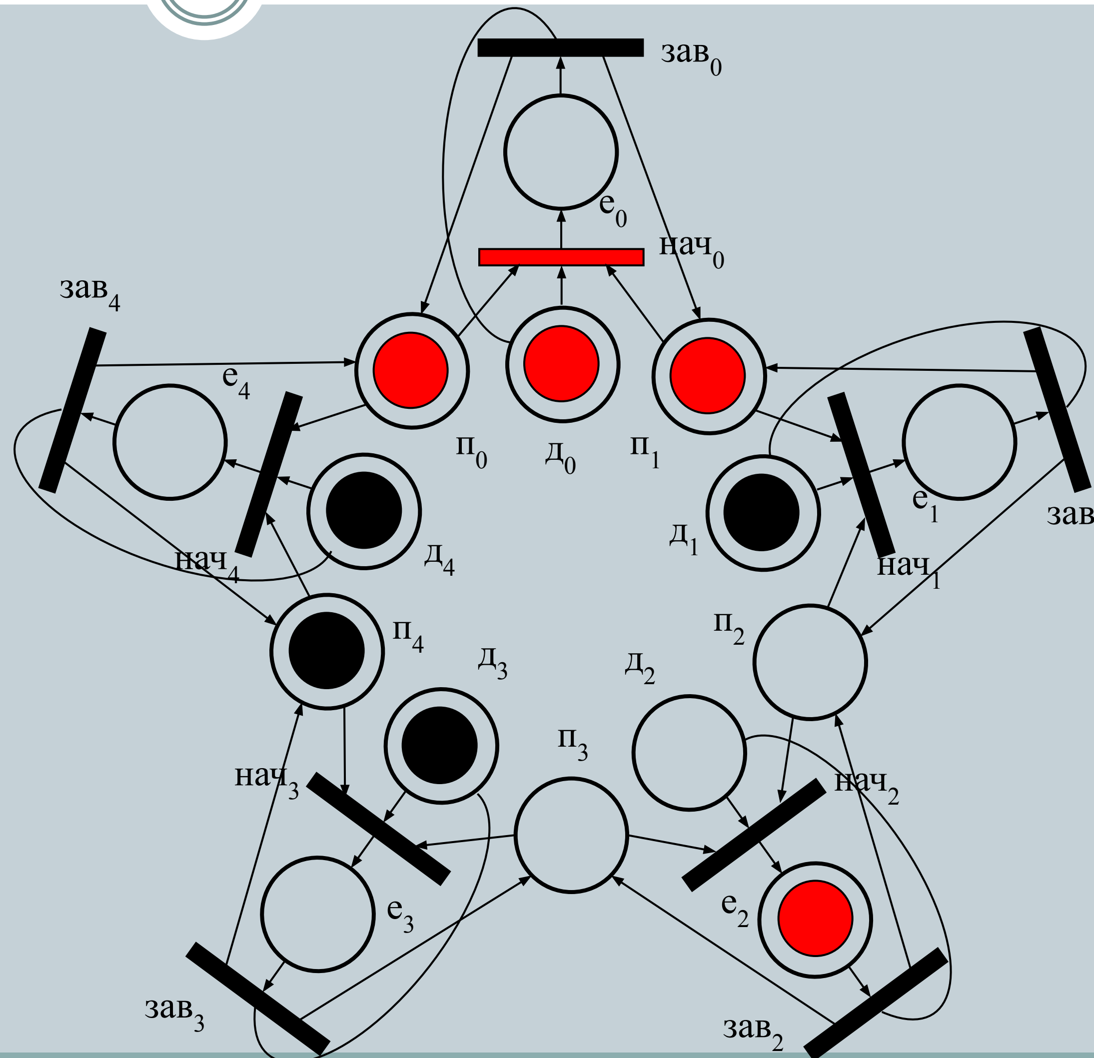
Моделирование систем

Задача об обедающих философах:



Моделирование систем

Задача об обедающих философах:



Моделирование систем

Задача об обедающих философах: