МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
імені Семена Кузнеця

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ

Л е к ц і я № 3

по курсу "**Операційні системи**"

Тема 2: Оперативна пам'ять, потоки і процеси ОС

Тема лекції:
Процеси та потоки в операційних системах

Лектор:
Доцент кафедри Інформаційних систем
кандидат технічних наук, доцент
Голубничий Дмитро Юрійович

18.09.2022

НАВЧАЛЬНІ ПИТАННЯ:

1. Процеси.
2. Потоки.
3. Синхронізація потоків в режимі ядра.
4. Синхронізація потоків в режимі користувача

1. Процеси

Процес *(process)* - абстракція, що описує поточну програму

Процес (process) - це програма та системні ресурси, необхідні для її роботи.

Процес - програма, яка перебуває в режимі виконання.

Процес *в Win32, Win64* - це об'єкт, який не виконується, а просто "володіє" виділеним йому адресним простором, іншими словами процес є структурою в пам'яті.

У виконавчій системі ***процеси*** - це об'єкти, що створюються і знищуються диспетчером об'єктів

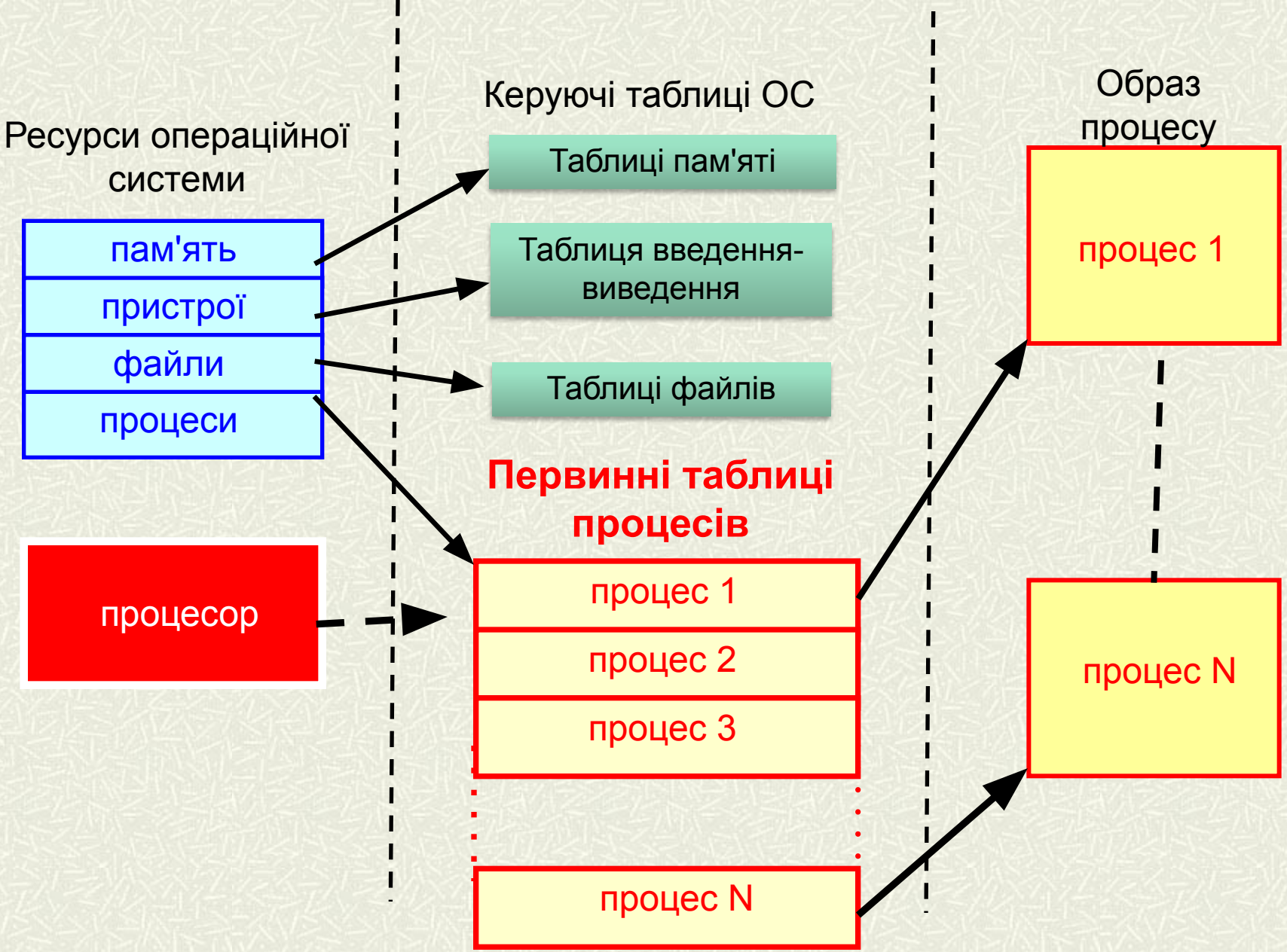
СКЛАДОВІ ПРОЦЕСУ

Виконавча програма (exe-модуль)

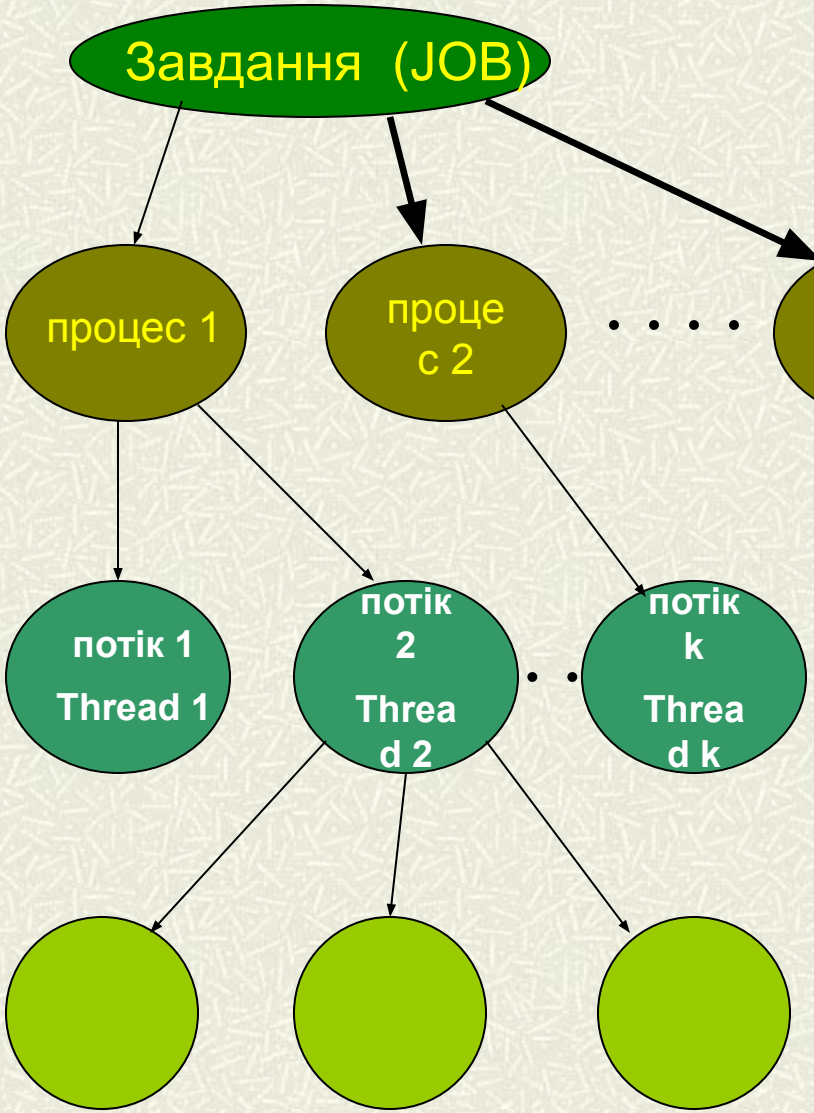
Закритий адресний простір

Системні ресурси

Потік управління



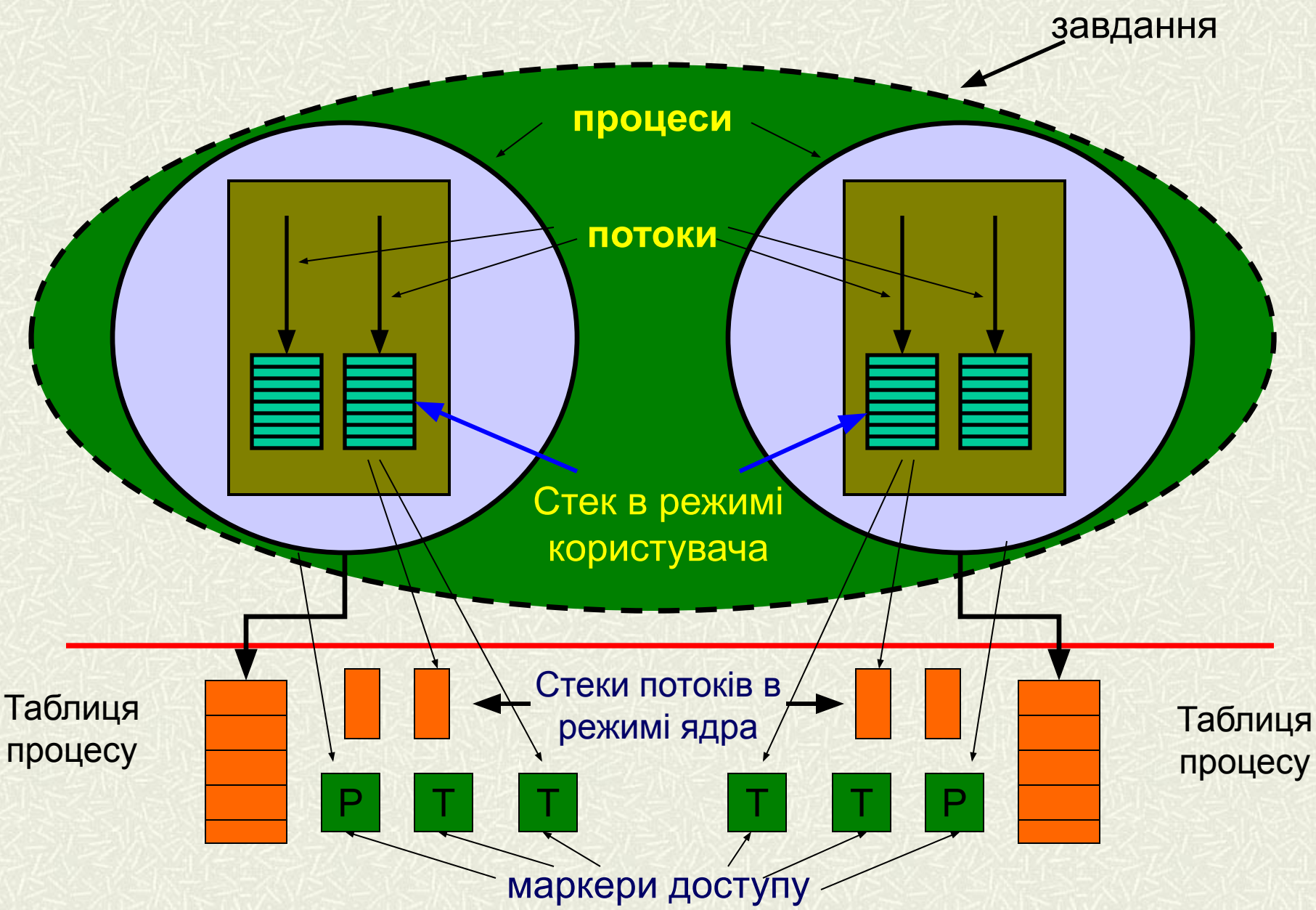
Концепція процесів і потоків



Волокна (Fibers)

об'єкти

Назва	Опис
Завдання	Набір процесів із загальними квотами і лімітами
Процес	Контейнер для ресурсів і потоків
Потік	Виконання коду в процесі
Волокно	Полегшений потік, повністю керований в просторі користувача



Взаємозв'язок між завданнями, процесами і потоками

Процес та його ресурси (приклад)

Тип об'єкта	Процес
Атрибути тіла об'єкта	Ідентифікатор процесу Маркер доступу Базовий пріоритет Процесорний спорідненість за замовчуванням Розміри квот Час виконання Лічильник введення / виведення Лічильник операцій віртуальної пам'яті Порти виключень-налагодження Коди завершення
Сервіси	Створити процес Відкрити процес Запросити інформацію процесу Встановити інформацію процесу Поточний процес Завершити процес Виділити-звільнити віртуальну пам'ять Читання / запис в віртуальну пам'ять Захистити віртуальну пам'ять Блокувати / розблокувати віртуальну пам'ять Опитати віртуальну пам'ять Скинути віртуальну пам'ять на диск

Об'єкт - процес

Атрибути об'єкта - процесу

Атрибут	призначення
Ідентифікатор процесу	Унікальне значення, яке ідентифікує процес в ОС
Маркер доступу	Об'єкт виконавчої системи, що містить інформацію про права зареєстрованого в системі користувача, якого представляє цей процес
Базовий пріоритет	Базовий пріоритет потоків процесу
Процесорна спорідненість за замовчуванням	Набір процесорів, на яких потоки процесу можуть виконуватися за замовчуванням
Розміри квот	Максимальний обсяг резидентної і нерезидентної системної пам'яті, простору в файлі підкачки і процесорного часу, що виділяється для користувача процесу
Час виконання	Загальний час виконання всіх потоків процесу
Лічильники введення-виведення	Змінні, в яких записується число і тип операцій введення-виведення, виконаних потоками процесу
Лічильники операцій віртуальної пам'яті	Змінні, в яких записується число і тип операцій віртуальної пам'яті, виконаних потоками процесу
Порти виключень-налагодження	Канали комунікацій між процесами, за якими диспетчер процесів посилає повідомлення, якщо один з потоків процесу викликає виняток
Код завершення	Причина завершення процесу

Інформація	Опис
Призначена для користувача програма	програма (exe-файл), яку потрібно виконати
Данні користувача	Змінна частина користувацького адресного простору (дані програми, призначений для користувача стек і модифікуємий код)
Системний стек	Один або кілька системних стеків для зберігання параметрів і адрес виклику процедур і системних служб
Керуючий блок процесу	Дані, необхідні ОС для управління процесом: 1) дескриптор процесу, 2) контекст процесу

Образ процесу: програма, дані, стек і атрибути процесу

1. ДЕСКРИПТОР ПРОЦЕСУ МІСТИТЬ

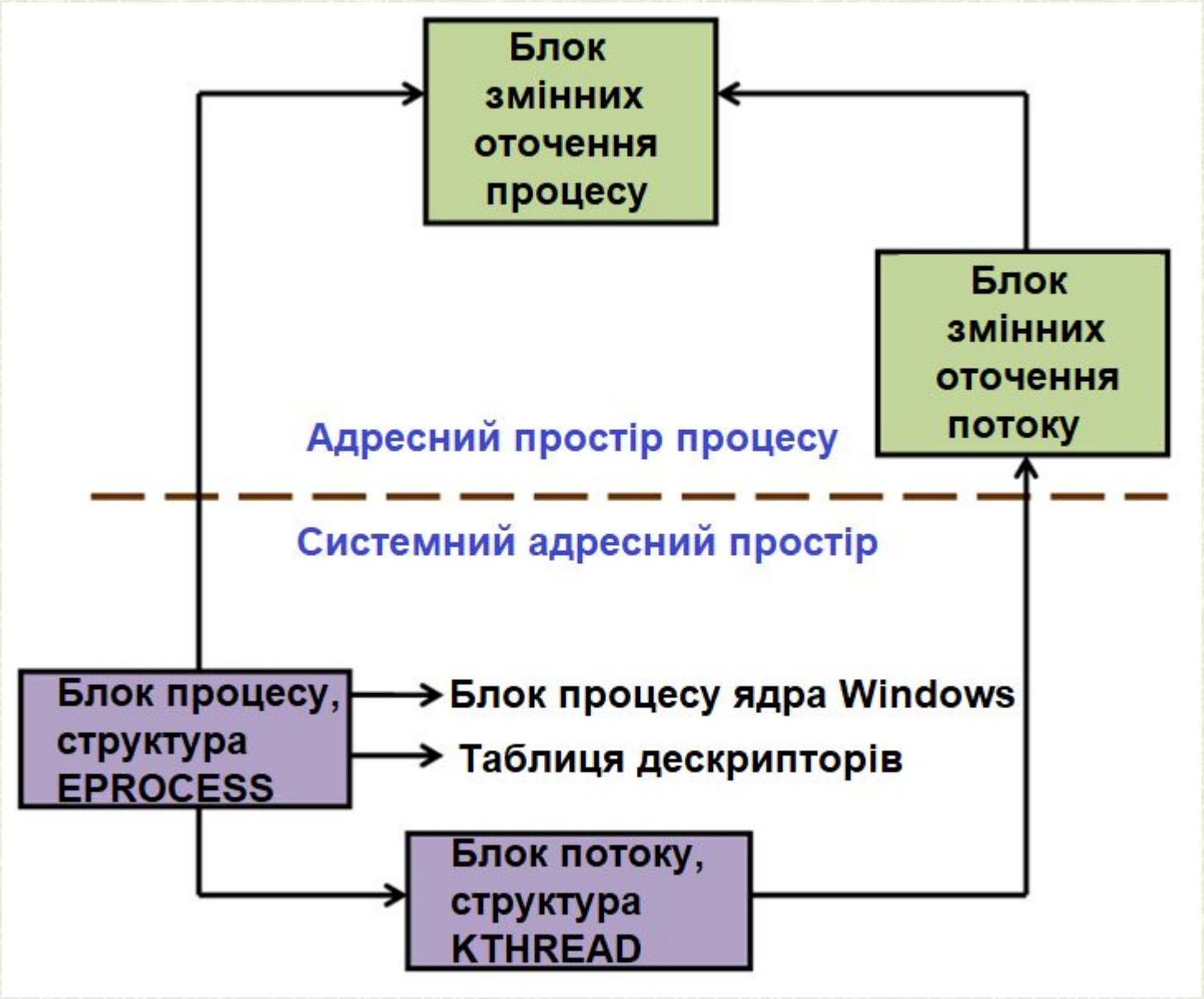
1. Інформацію щодо ідентифікації процесу (ідентифікатор процесу, ідентифікатор користувача, ідентифікатори батьківського і дочірніх процесів).
2. Інформацію про стан процесу
3. Інформацію, яку використовують для управління процесом

2. КОНТЕКСТ ПРОЦЕСУ

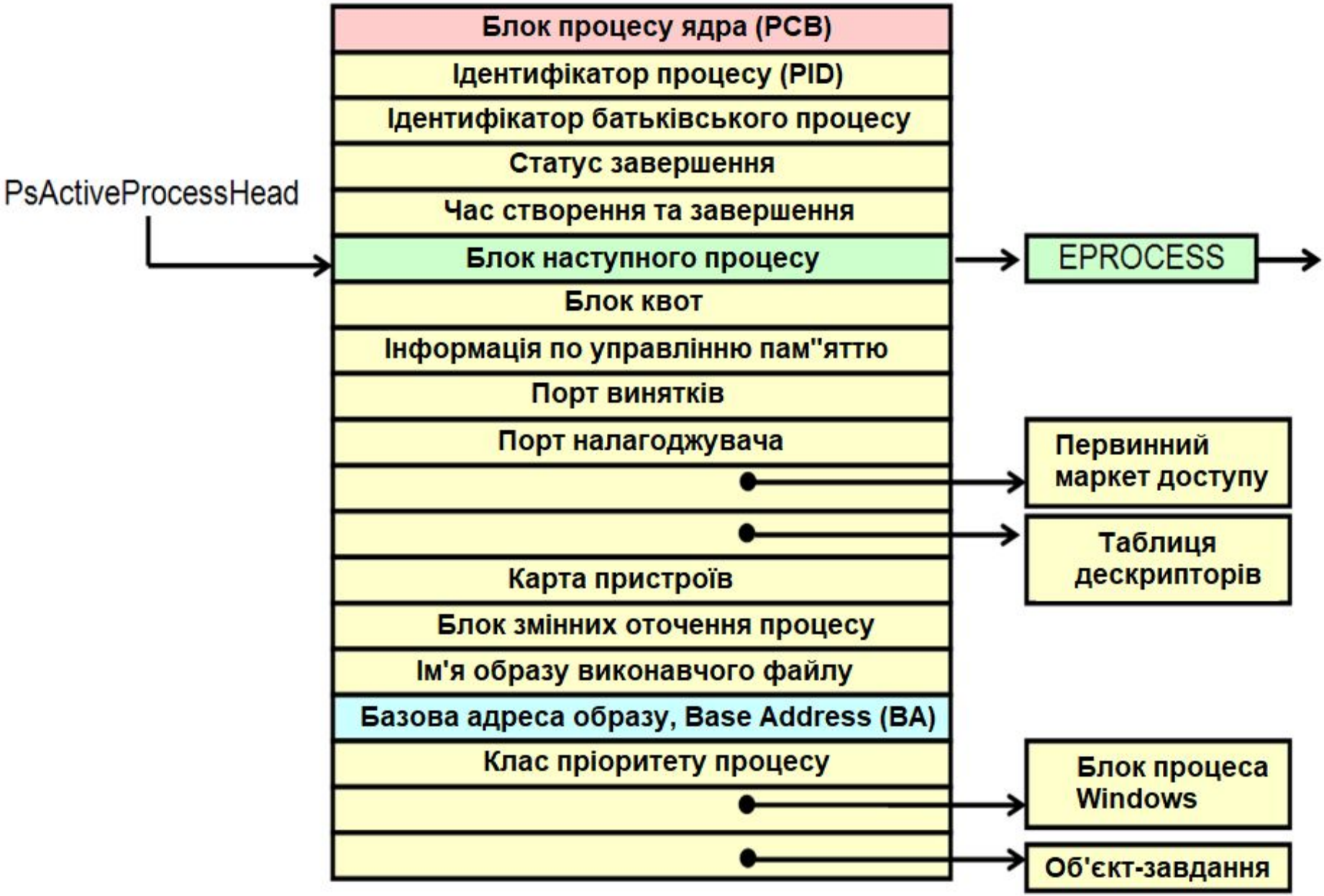
1. Вміст регістрів процесора, доступних користувачеві (зазвичай 8 - 32 регістра і до 100 регістрів в RISC - процесорах);
2. Вміст лічильника команд;
3. Стан регістрів і регістрів стану;
4. Коди умови, що відображають результат виконання останньої арифметичної або логічної операції (наприклад, рівність нулю, переповнення тощо);
5. Покажчики вершин стеків, що зберігають параметри і адреси виклику процедур і системних служб.

1. Стан процесу, що визначає його готовність до виконання (що виконується, готовий до виконання, що очікує події, який був загальмований);
2. Дані про пріоритет (поточний, за замовчуванням, максимально можливий);
3. Інформація про події - ідентифікація події, настання якого дозволить продовжити виконання процесу;
4. Покажчики, що дозволяють визначити розташування образу процесу в оперативній пам'яті і на диску;
5. Покажчики на інші процеси (що знаходяться в черзі на виконання);
6. Прапори, сигнали і повідомлення, що мають відношення до обміну інформацією між двома незалежними процесами;
7. Дані про привілеї, що визначають прав доступу до певної області пам'яті або можливості виконувати певні види команд, використовувати системні утиліти і служби;
8. Покажчики на ресурси, якими управляє процес;
9. Відомості щодо використання ресурсів і процесора;
10. Інформація, пов'язана з плануванням.

Інформація про стан і управління процесом



Структури Windows для управління процесами і потоками (узагальнена схема)



Структура EPROCESS

Порядок пошуку файлу виконуваної програми

1. В каталозі, в який завантажено додаток

↓ якщо немає , то

2. У поточному каталозі

↓ якщо немає , то

3. У системному каталозі Windows (%SystemRoot%\)

↓ якщо немає , то

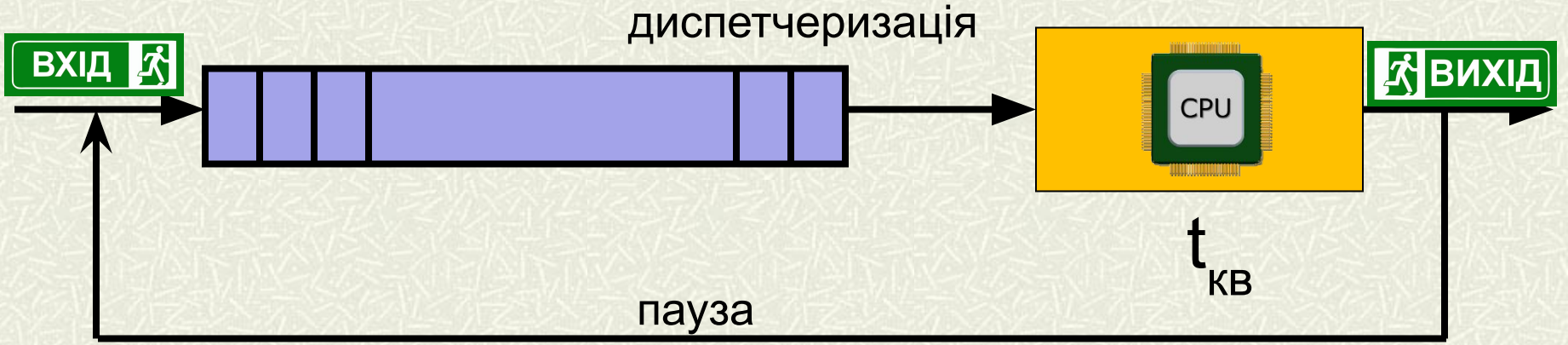
4. У каталозі Windows, який повертає функція GetWindowDirectory()

↓ якщо немає , то

5. У каталогах, перерахованих у змінній середовища PATH, яка визначає порядок пошуку в них

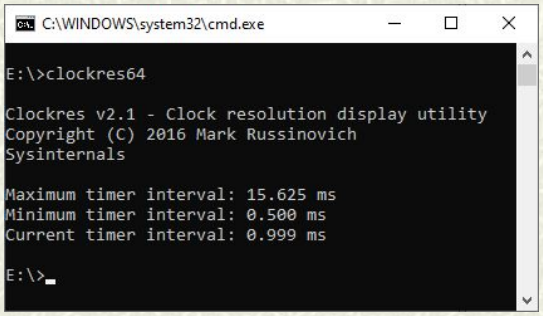
Проста модель процесу

Граф станів і переходів

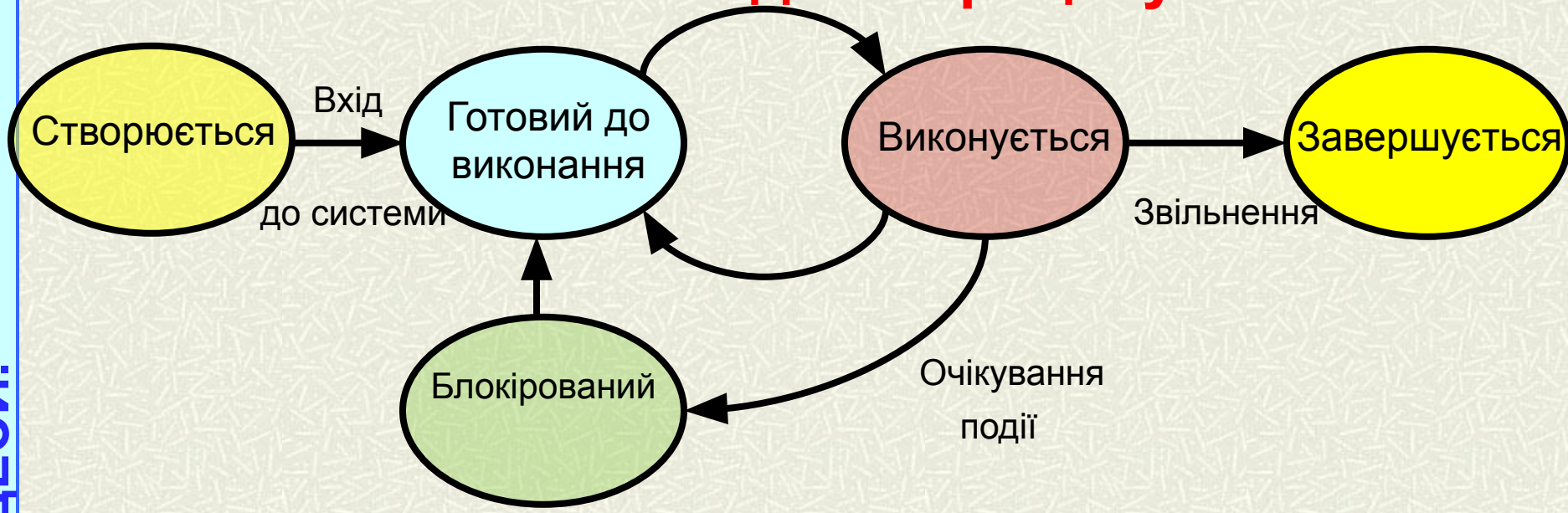


Квантом процесорного часу називається час, після якого виконання потоку з алгоритмом планування SCHED_RR буде перервано, а управління буде передано іншому потоку з тим самим пріоритетом.

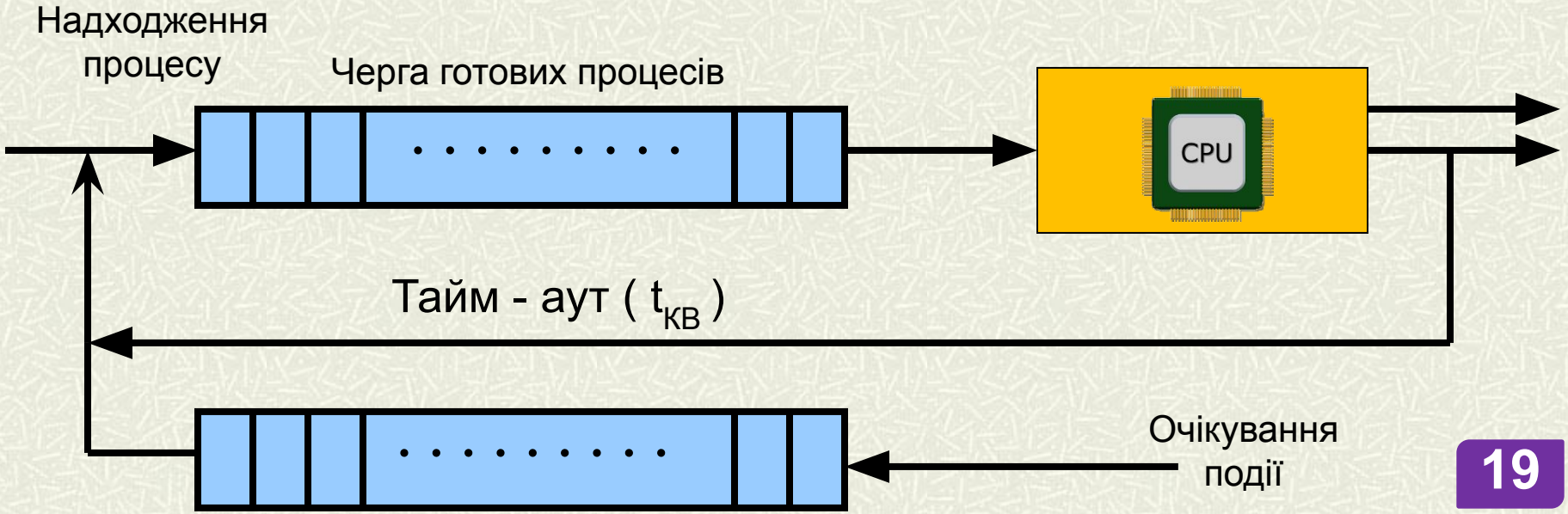
tкв можливо виміряти утилітою Clockres від sysinternals.com



Загальна модель процесу

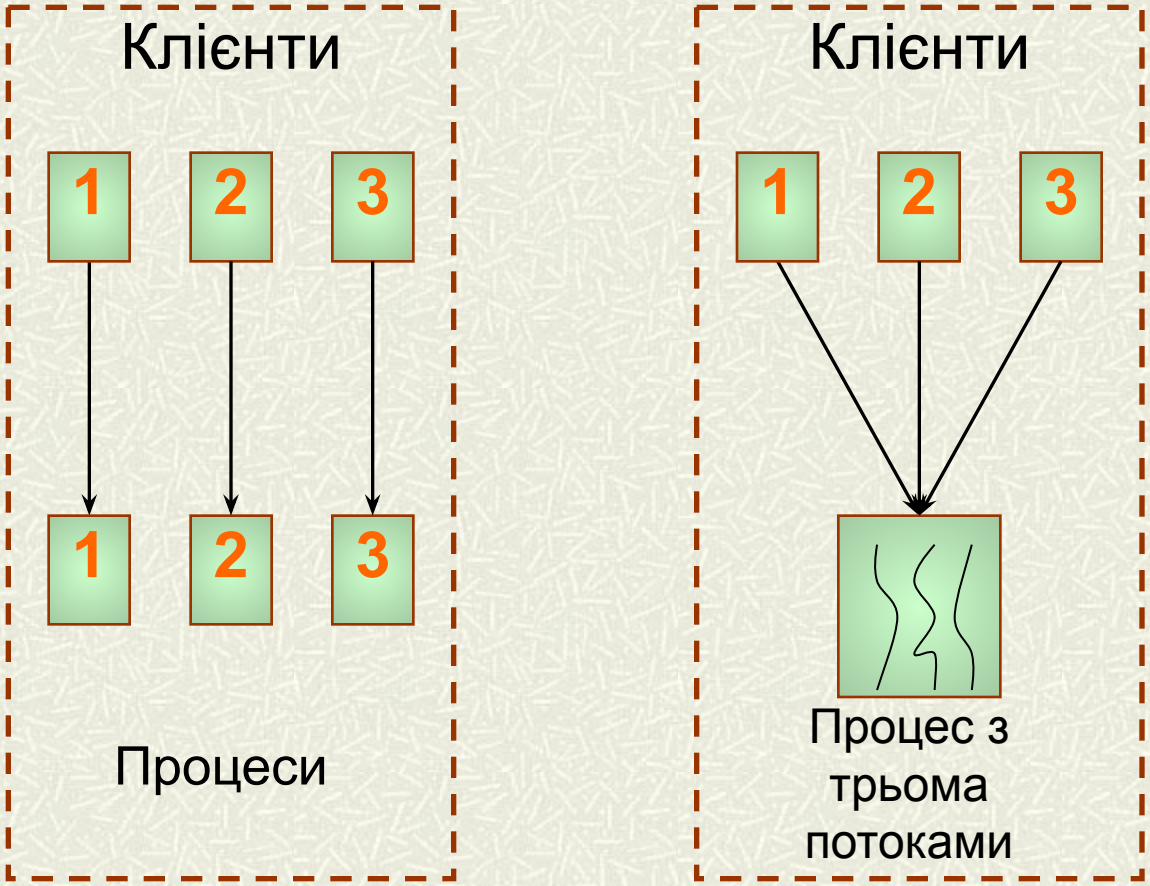


Граф станів і переходів



2. Потоки

Потік (Thread) **– це незалежний шлях виконання всередині процесу, що розділяє разом з процесом загальний адресний простір, код і глобальні дані.**



Переваги використання потоків:

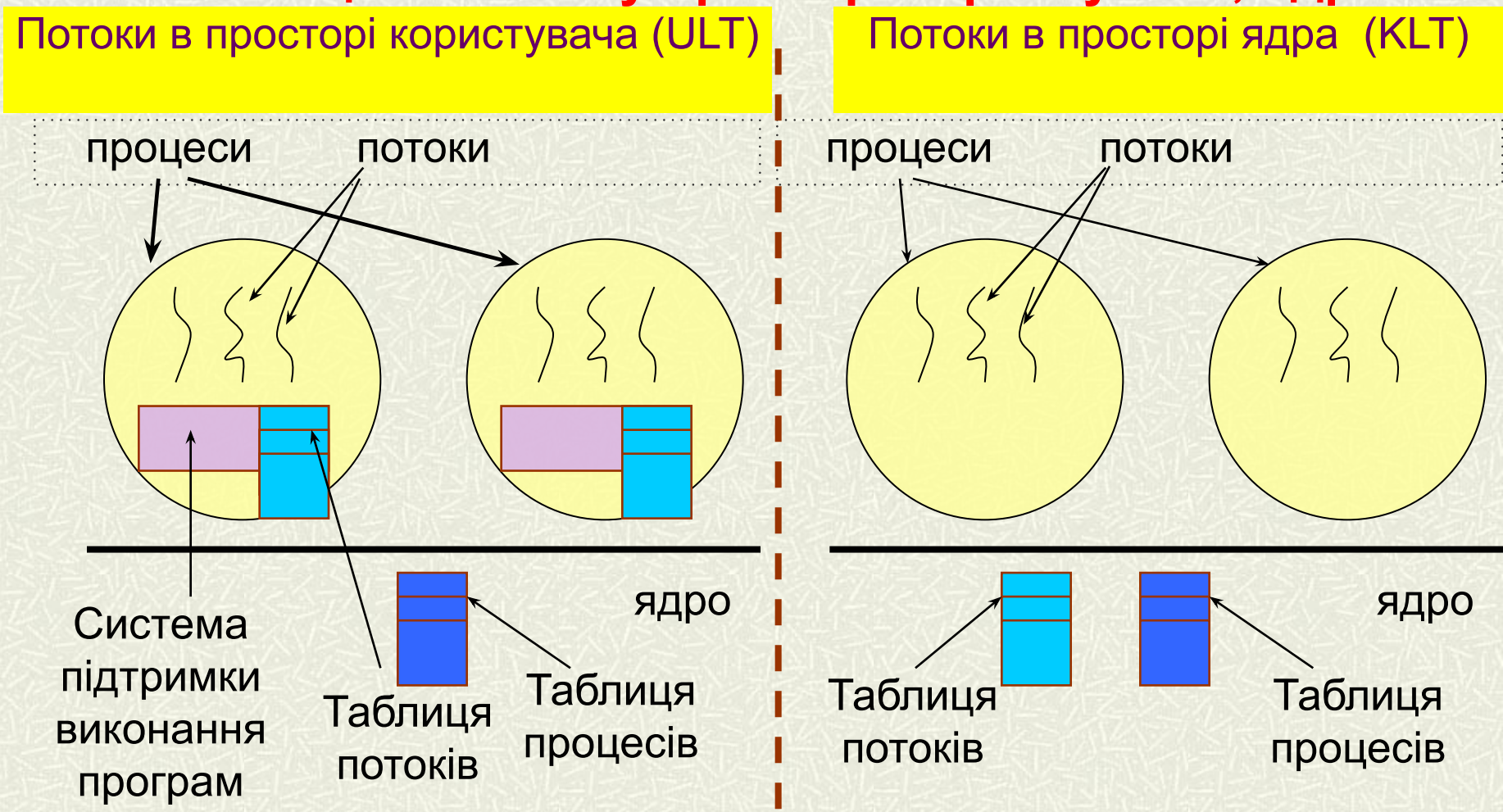
1. Спрощення програми

2. Швидкість створення потоку

3. Підвищення продуктивності

Порівняння багатопотокової системи з однопоточній

Реалізація потоків у просторі користувача, ядра



Способи реалізації пакету потоків:

- в просторі користувача (user - level threads - ULT);
- в ядрі (kernel - level threads - KLT).

ПОТІК НА РІВНІ КОРИСТУВАЧА (ULT)

ПЕРЕВАГИ:

- можна реалізувати в ОС, що не підтримує потоки без будь-яких змін в ОС;
- висока продуктивність, оскільки процесу не потрібно перемикатися в режим ядра і назад;
- ядро про потоках нічого не знає і управляє однопоточними процесами;
- є можливість використання будь-яких алгоритмів планування потоків з урахуванням їх специфіки;
- управління потоками покладається на програму користувача.

НЕДОЛІКИ:

- системний виклик блокує не тільки працюючий потік, але і всі потоки того процесу, до якого він відноситься;
- додаток не може працювати в багатопроцесорному режимі, так як ядро закріплює за кожним процесом тільки один процесор;
- при запуску одного потоку жоден інший потік в рамках одного процесу не буде запущений поки перший добровільно не віддає процесор;
- всередині одного потоку немає переривань за таймером, в результаті чого неможливо створити планувальник за таймером для почергового виконання потоків.

ПОТІК НА РІВНІ ЯДРА (KLT)

ПЕРЕВАГИ:

- можливо планування роботи декількох потоків одного і того ж процесу на декількох процесорах;
- реалізується мультипрограмування в рамках всіх процесів (у тому числі одного);
- при блокуванні одного з потоків процесу ядро може вибрати інший потік цього ж (або іншого процесу);
- процедури ядра можуть бути багатопоточними

НЕДОЛІКИ:

- необхідність дворазового перемикання режиму користувач - ядро, ядро - користувач для передачі управління від одного потоку до іншого в рамках одного і того ж процесу.

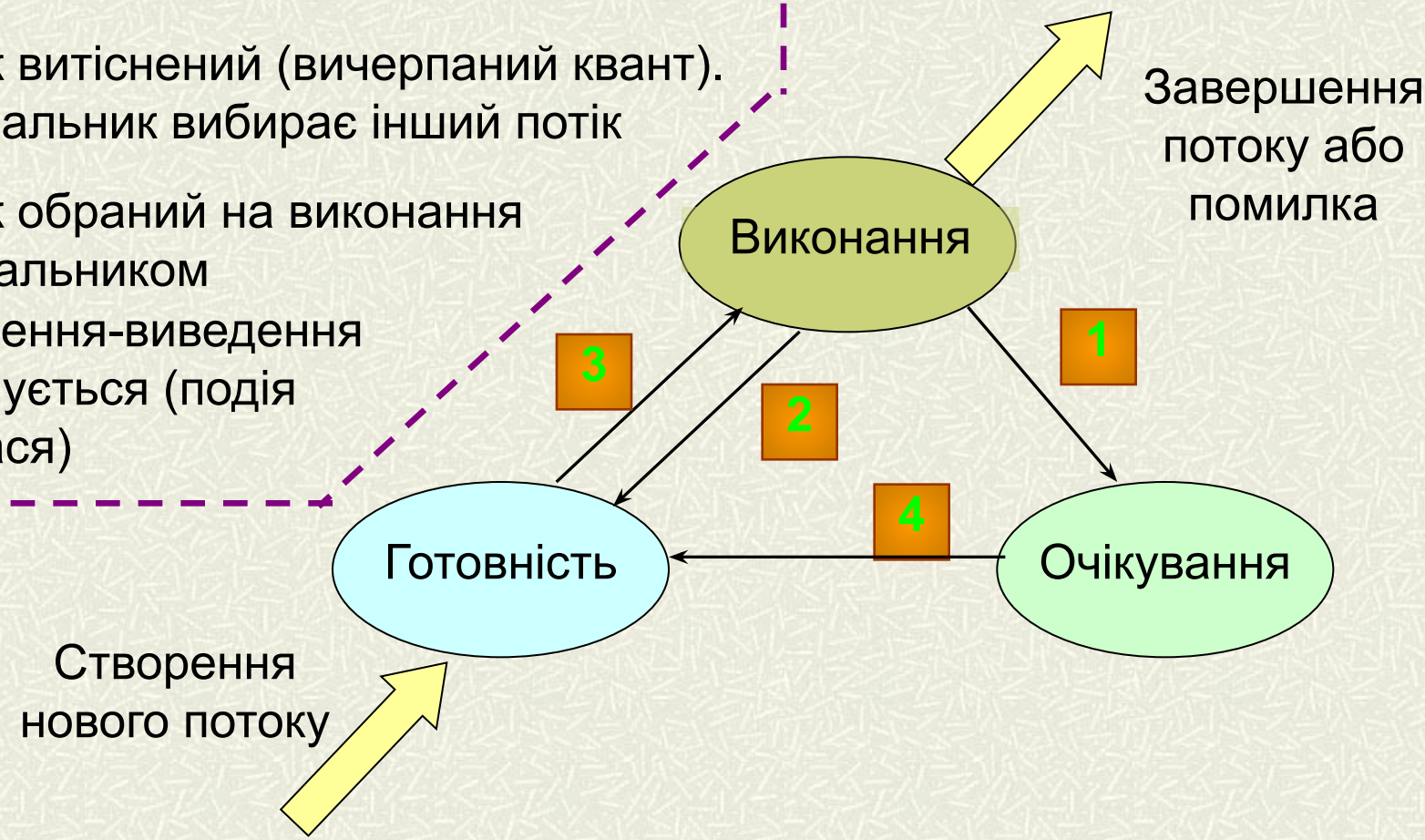
Стан потоків

1. Потік блокується, чекаючи введення-виведення даних або іншої події

2. Потік витіснений (вичерпаний квант). Планувальник вибирає інший потік

3. Потік обраний на виконання планувальником

4. Введення-виведення завершується (подія відбулася)



Типовий граф стану потоку

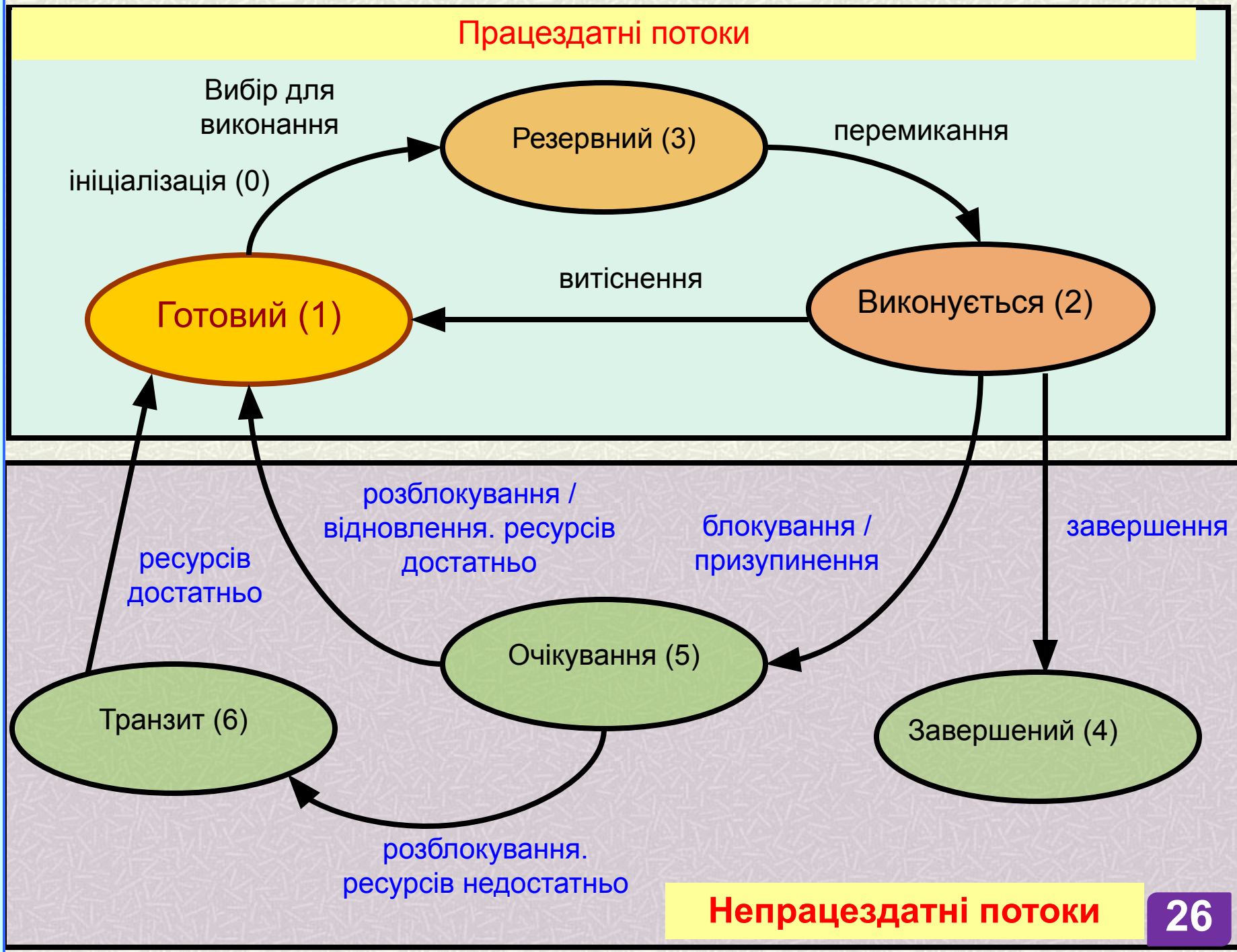

Працездатні потоки
Вибір для виконання
Резервний (3)
перемикання
ініціалізація (0)
витіснення
Готовий (1)
Виконується (2)
розблокування / відновлення. ресурсів достатньо
блокування / призупинення
завершення
ресурсів достатньо
Очікування (5)
Транзит (6)
Завершений (4)
розблокування. ресурсів недостатньо
Непрацездатні потоки

СХЕМА ПЛАНУВАННЯ З УРАХУВАННЯМ ЧЕРГ ЗАВДАНЬ (ПОТОКОВ)

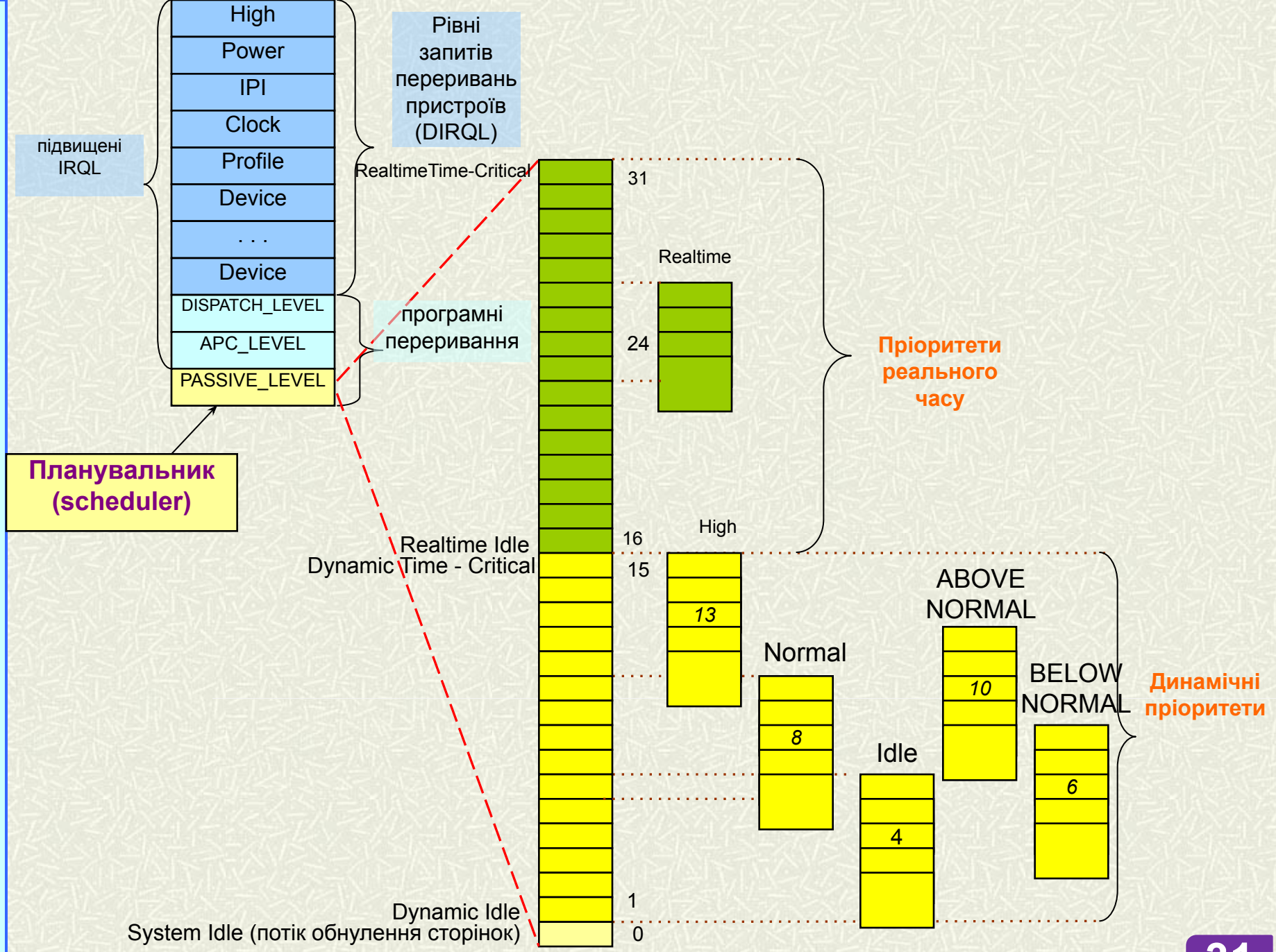
Тип об'єкта	потік
Атрибути тіла об'єкта	Клієнтський ідентифікатор Контекст потоку Базовий пріоритет Динамічний пріоритет Процесорний спорідненість потоку Час виконання потоку Стан тривоги Лічильник припинень Маркер імперсонації Порти завершення Код завершення
Сервіси	Створити потік Відкрити потік Запросити інформацію потоку Встановити інформацію потоку Поточний потік Завершити потік Запросити контекст Встановити контекст Призупинити Відновити виконання Тривога потоку Перевірити стан тривоги потоку Зареєструвати порт завершення

Об'єкт - потік

Атрибути об'єкта-потоку

Атрибут	призначення
Ідентифікатор клієнта	Унікальне значення, яке ідентифікує потік при виклику їм сервера
Контекст потоку	Набір значень регістрів і інших непостійних даних, що визначають стан виконання потоку
Базовий пріоритет	Нижня межа динамічного пріоритету потоку
Динамічний пріоритет	Пріоритет потоку на даний момент
Процесорний спорідненість потоку	Набір процесорів, на яких може виконуватися потік, (неповне) підмножина процесорного спорідненості процесу потоку
Час виконання потоку	Загальний час виконання потоку в призначеному для користувача режимі і режимі ядра
Статус оповіщення	Прапор, який вказує на необхідність відпрацювання потоком асинхронного виклику процедури (APC)
Лічильник припинень	Кількість припинень виконання потоку без подальшого відновлення
Маркер імперсонації	Тимчасовий маркер доступу, що дозволяє потоку виконувати дії від імені іншого процесу (використовується підсистемами)
Порт завершення	Канал комунікації між процесами, в який диспетчер процесів посилає повідомлення при завершенні потоку (використовується підсистемами)
Код завершення потоку	Причина завершення потоку

Пріоритети потоків



Система пріоритетів

ЗМІНА БАЗОВОГО ПРІОРИТЕТУ ПОТОКУ

збільшення пріоритету

+ 1 - завершення введення-виведення по диску;

+2 - для послідовної лінії;

+ 6 - клавіатура;

+ 8 - звукова карта;

+ 2 - знімається блокування по семафора (для потоку переднього плану);

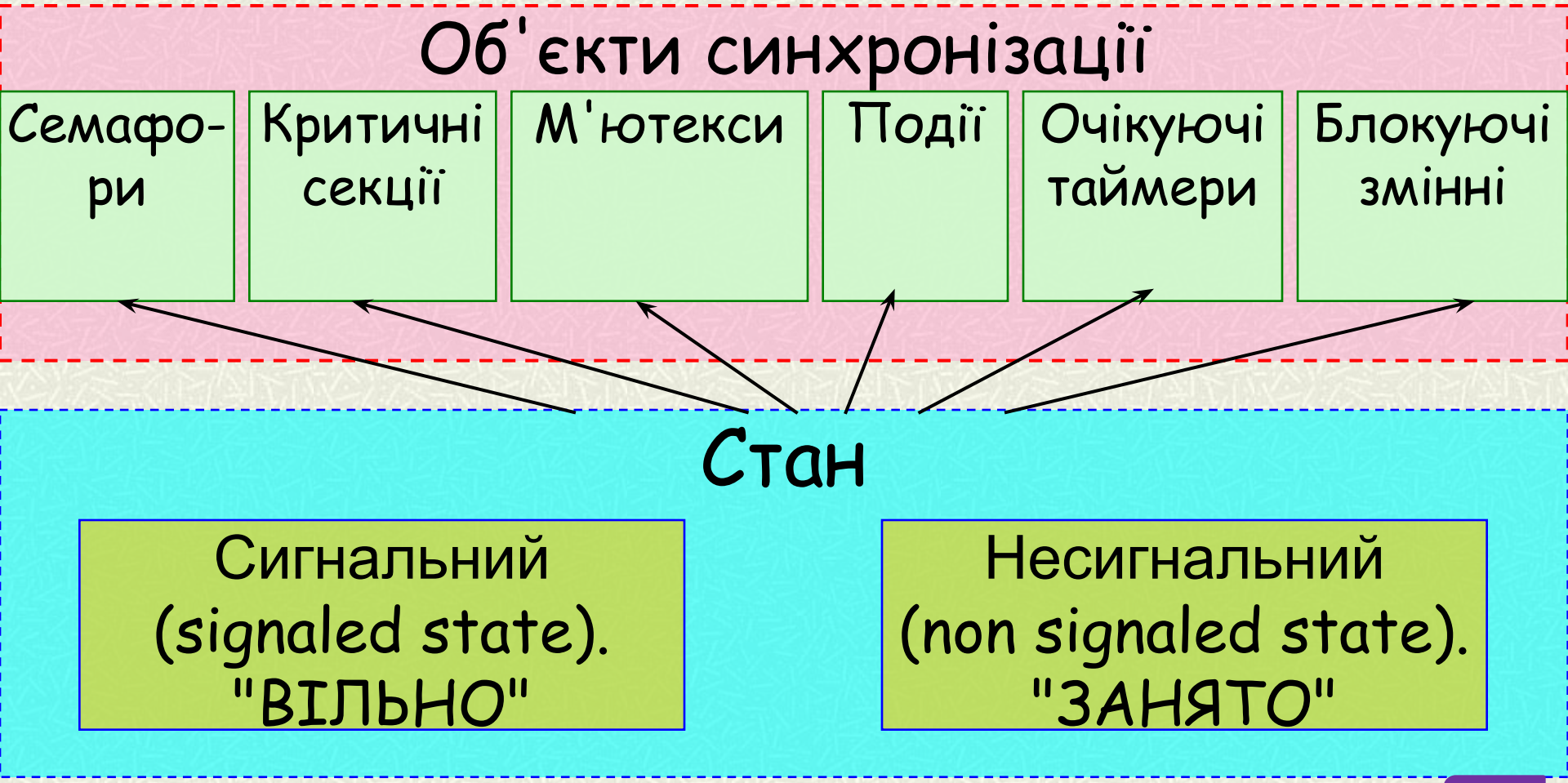
+ 1 - знімається блокування по семафора (для потоку непереднього плану); пріоритет 15 на 2 кванта процесора, якщо готовий до виконання потік простоює понад деякого директивного часу.

зменшення пріоритету

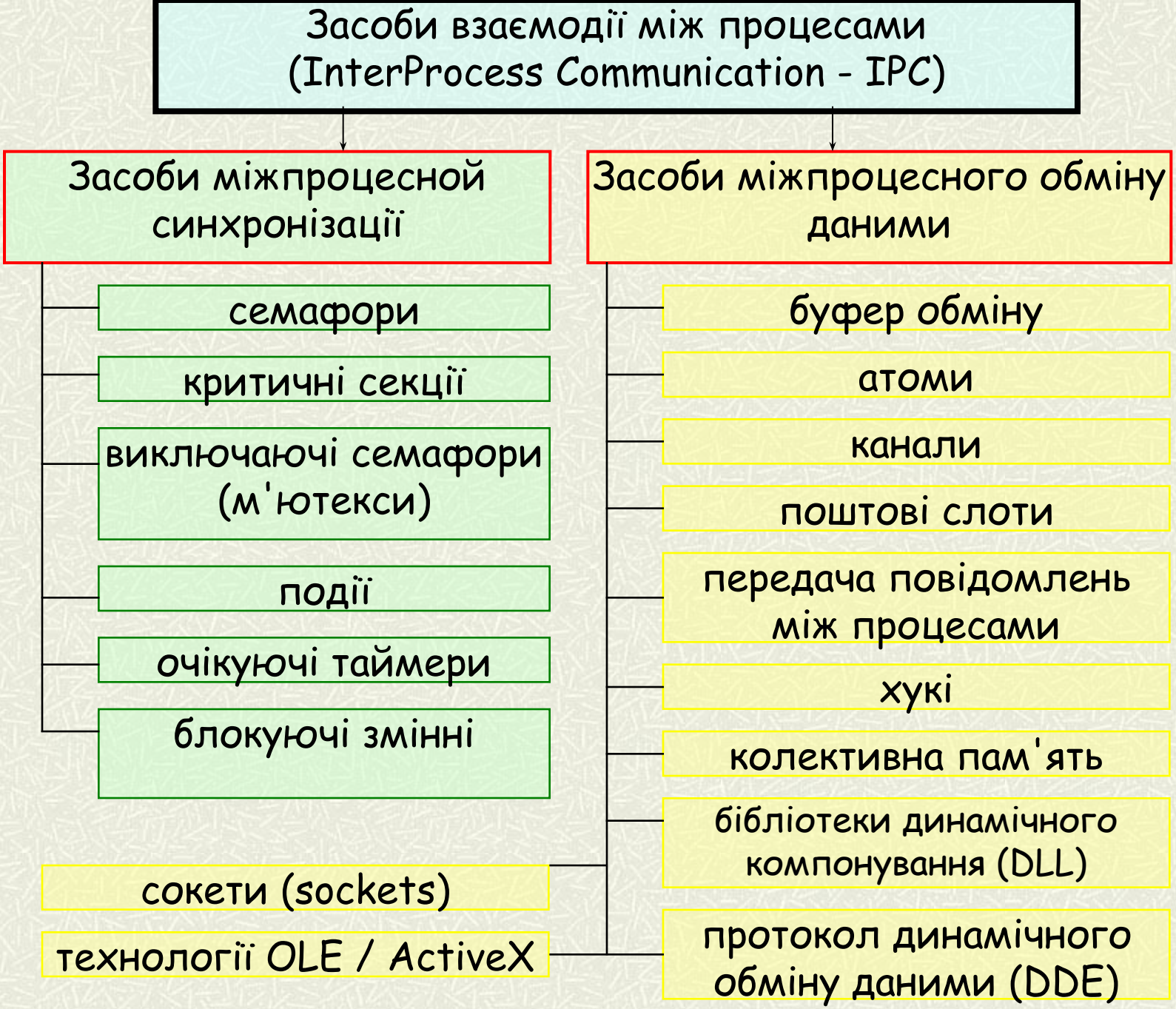
- 1 - якщо повністю використаний квант часу процесора (багаторазово, аж до базового пріоритету).

3. Синхронізація потоків в режимі ядра

СИНХРОНІЗАЦІЯ - узгодження швидкостей виконання потоків шляхом припинення потоку до настання деякої події і подальшій його активізації при настанні цієї події



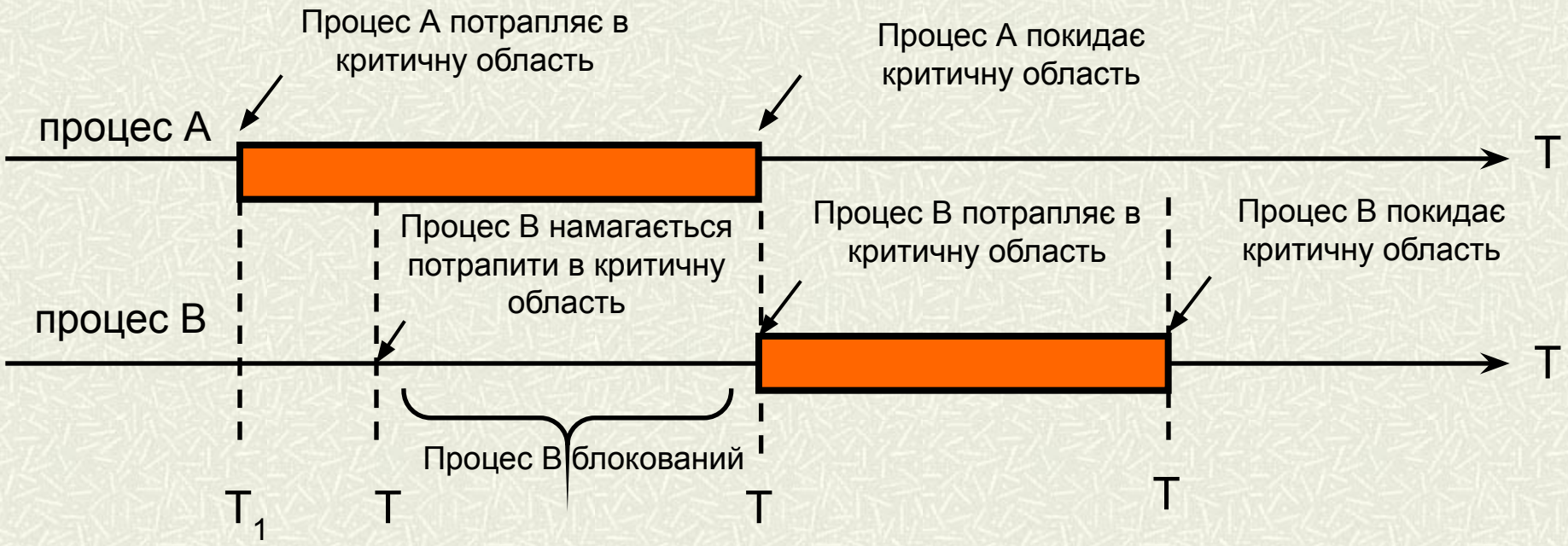
Види станів об'єктів синхронізації



Класифікація засобів взаємодії між процесами

Конкуренція процесів в боротьбі за ресурси

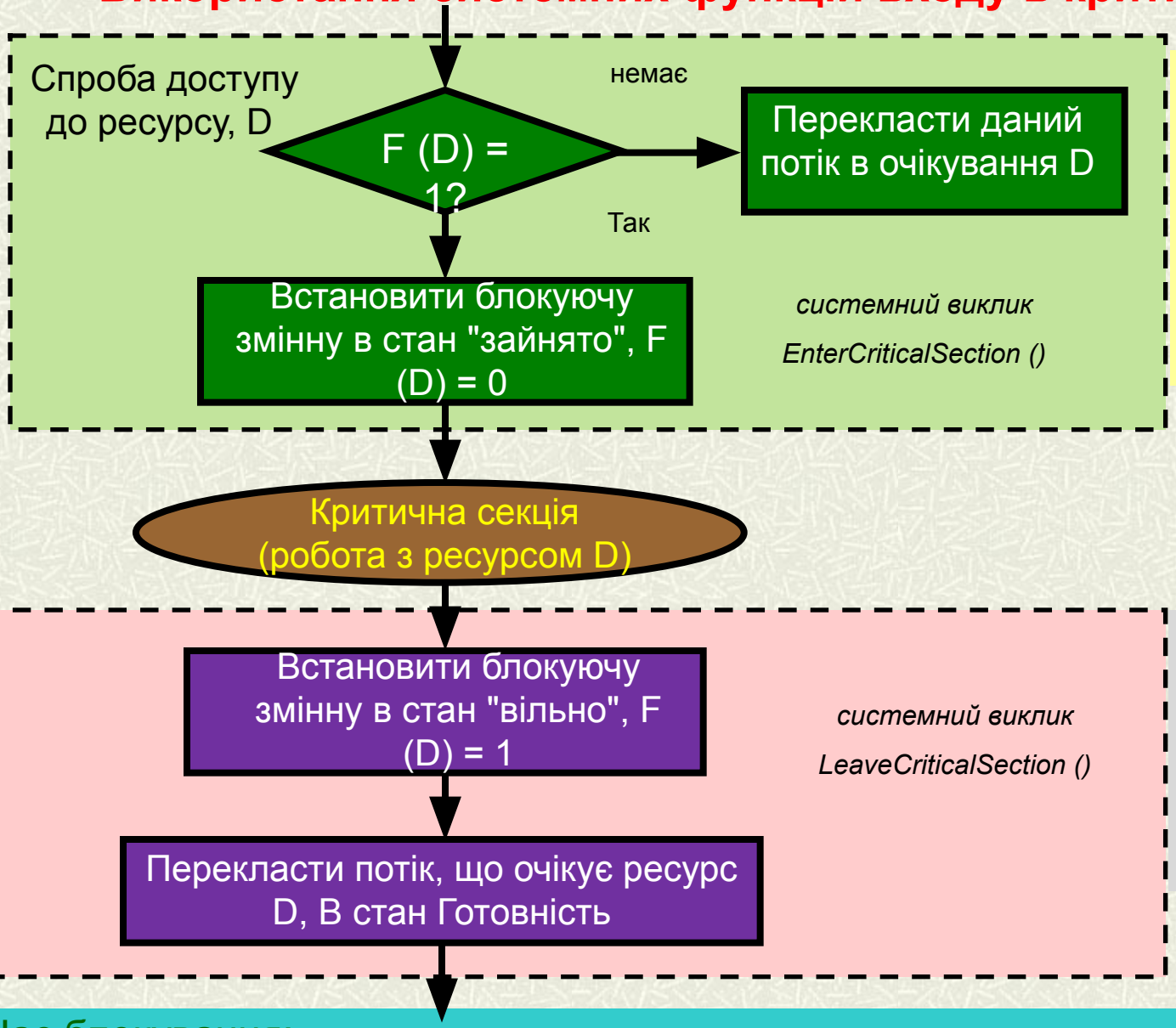
Конкуренція - ситуація, коли два або більше процесів вимагають доступ до одного і того ж ресурсу (принтеру, файлу і т.п.), званому критичним.



необхідність взаємовиключення:

1. Процеси не повинні одночасно перебувати в критичних областях.
2. У програмі не повинно бути припущень про швидкість або кількості процесів.
3. Процес, що знаходиться поза критичної області, не може блокувати інші процеси.
4. Неможлива ситуація, в якій процес вічно чекає попадання в критичну область.

Використання системних функцій входу в критичну секцію



Перевага:
виключається втрата часу процесора на циклічну перевірку звільнення зайнятого ресурсу.

Недолік:
ростуть накладні витрати ОС на по реалізації функції входу в критичну секцію і виходу з неї

Час блокування:
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session Manager
параметр *CriticalSectionTimeout = 2592000 секунд (30 діб)*

4. Сінхронізація потоків в режимі користувача

4.1. Семафори

Взаємоблокування

Умови виникнення взаимоблокуванням (тупикової) ситуації:

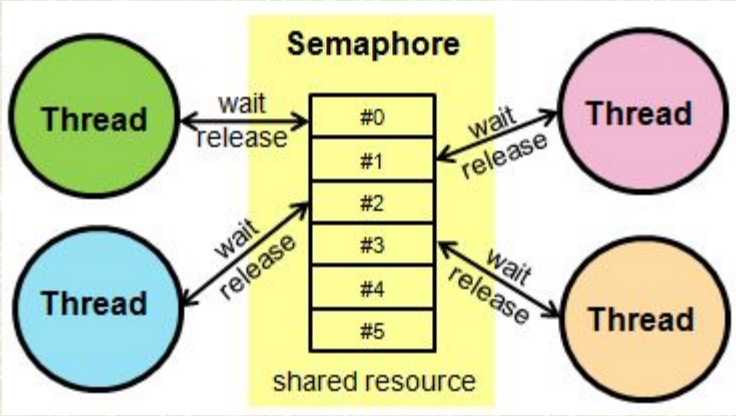
1. **Взаємне виключення.** Кожен ресурс в даний момент або відданий рівно одному процесу, або недоступний.
2. **Умова утримання і очікування.** Процеси, в даний момент утримують отримані раніше ресурси, можуть запитувати нові ресурси.
3. **Відсутність примусового вивантаження ресурсів.** У процесу не можна забрати примусово раніше отримані ресурси.
4. **Умова циклічного очікування.** Існує кругова послідовність з двох і більше процесів, кожен з яких чекає доступу до ресурсу, утримуваного наступним членом послідовності.

Стратегії боротьби з взаимоблокуванням:

1. Нехтування проблемою в цілому.
2. Виявлення та усунення взаємоблокувань (відновлення).
3. Недопущення тупикових ситуацій за допомогою акуратного розподілу ресурсів.
4. Запобігати за допомогою структурного спростування одного з чотирьох умов, необхідних для взаємоблокування.

СЕМАФОРИ ДІЙКСТРИ (DIJKSTRA)

Семафор — це універсальний механізм для організації взаємодії процесів (в термінології операційних систем сімейства Windows — потоків). Визначення семафору зроблено нідерландським вченим Едсгером Дейкстрою, деякий час використовувався термін Семафор Дейкстри



Семафор: змінна **S**, примітиви **P** (Proberen - перевірка; down) і **V** (Verhogen - збільшення, up)

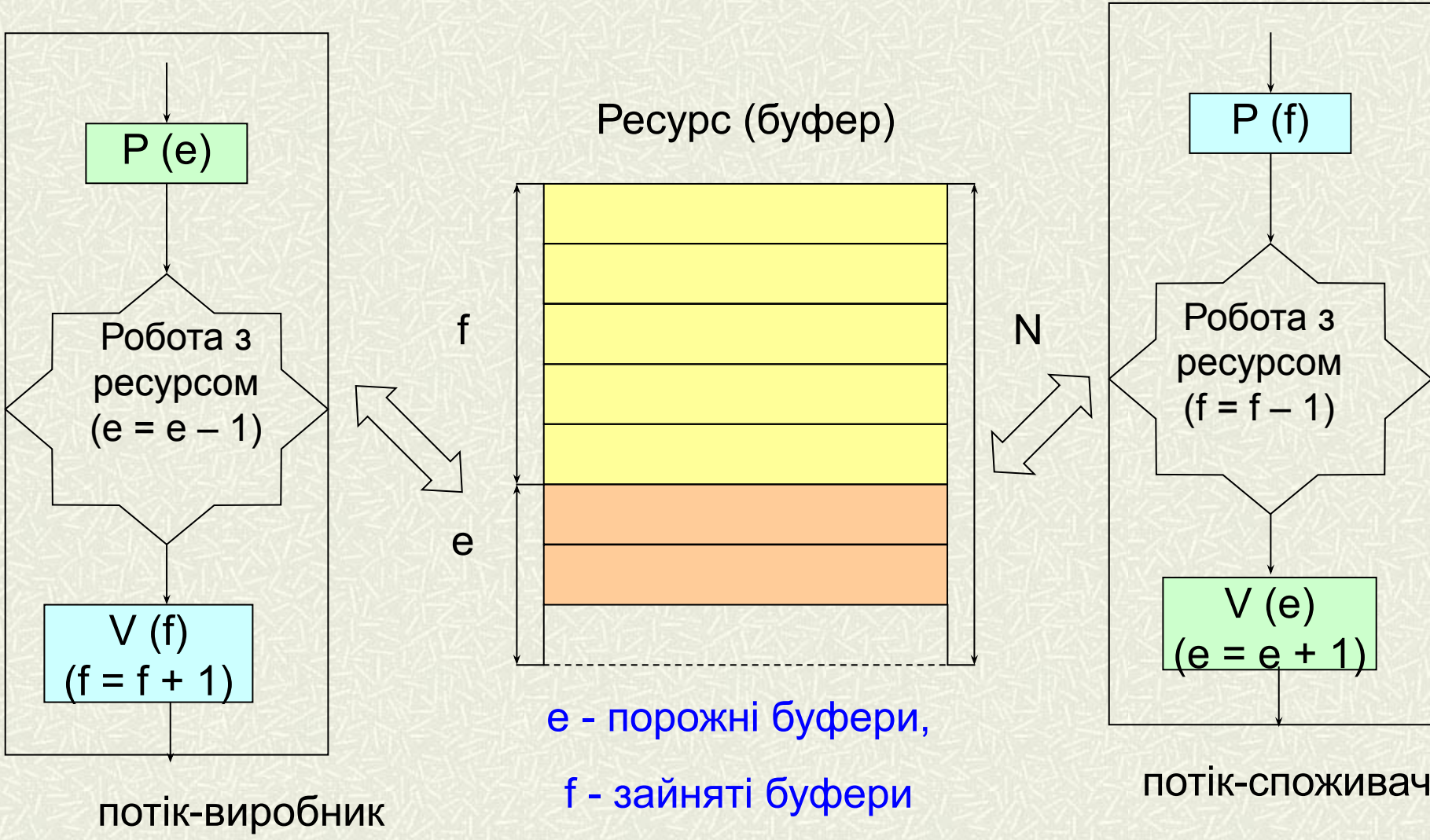
V (S) - змінна S збільшується на 1 єдиним дією. Вибірка, нарощування і запам'ятовування не можуть бути перервані. До змінної S немає доступу під час виконання цієї операції.

P (S) - змінна S зменшується на 1, Якщо це можливо, складе в області невід'ємних значень. Якщо S зменшити неможливо, потік, що виконує операцію P, чекає, поки це зменшення стане можливим. Операція P неподільна.

В окремому випадку семафор S може приймати виконавчі значення 0 і 1, перетворюючись в блокуючу змінну (двійковий семафор).

Операція P укладає в собі потенційну можливість переходу процесу, який її виконує, в стан очікування (якщо S = 0).

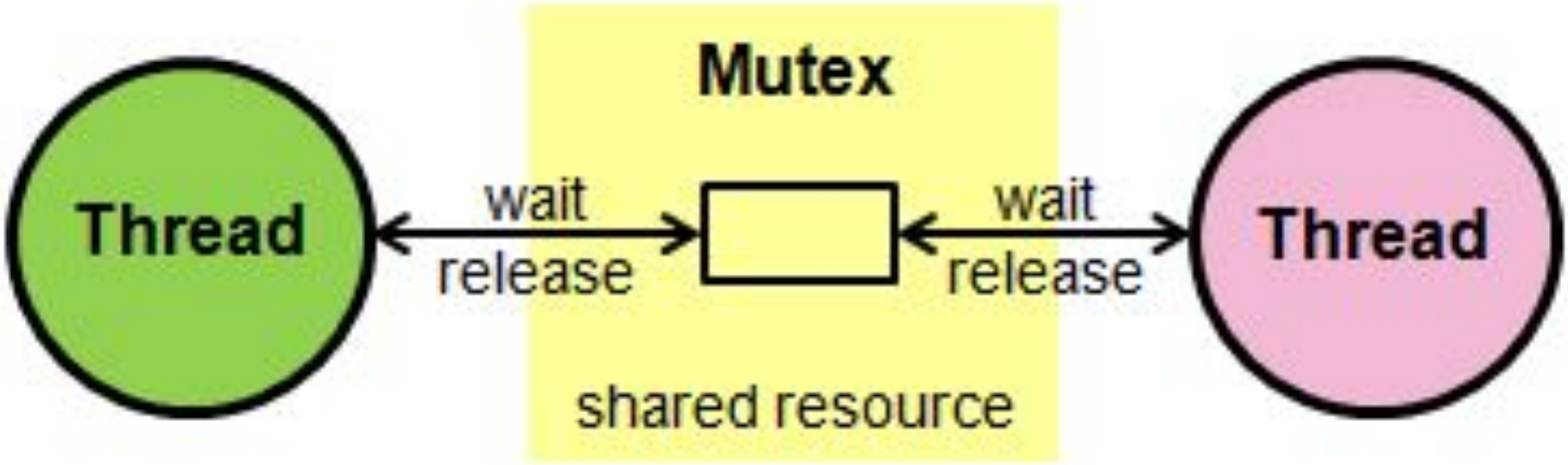
Початкові значення семафорів: e = N
f = 0



4. Сінхронізація потоків в режимі користувача

4.2. М'ютекси і події

М'ютекс (mutex, **mut**ual **ex**clusion — взаємне виключення) призначено для захисту певного об’єкта у потоці від доступу інших потоків. М'ютекс є одним із засобів синхронізації роботи потоків або процесів



М'ютекси — ***це прості двійкові семафори, які можуть перебувати в одному з двох станів*** - сигнальному або несигнальному (відкритий і закритий відповідно). Коли потік отримує м'ютекс, той переводиться в несигнальний стан.

Організація послідовного доступу до ресурсів з використанням м'ютексів стає нескладною, оскільки в кожен конкретний момент тільки один потік може володіти цим об'єктом. Для того, щоб об'єкт mutex став доступний потокам, що належать різним процесам, при створенні йому необхідно присвоїти ім'я. Потім це ім'я потрібно передати «у спадок» завданням, які повинні його використовувати для взаємодії.

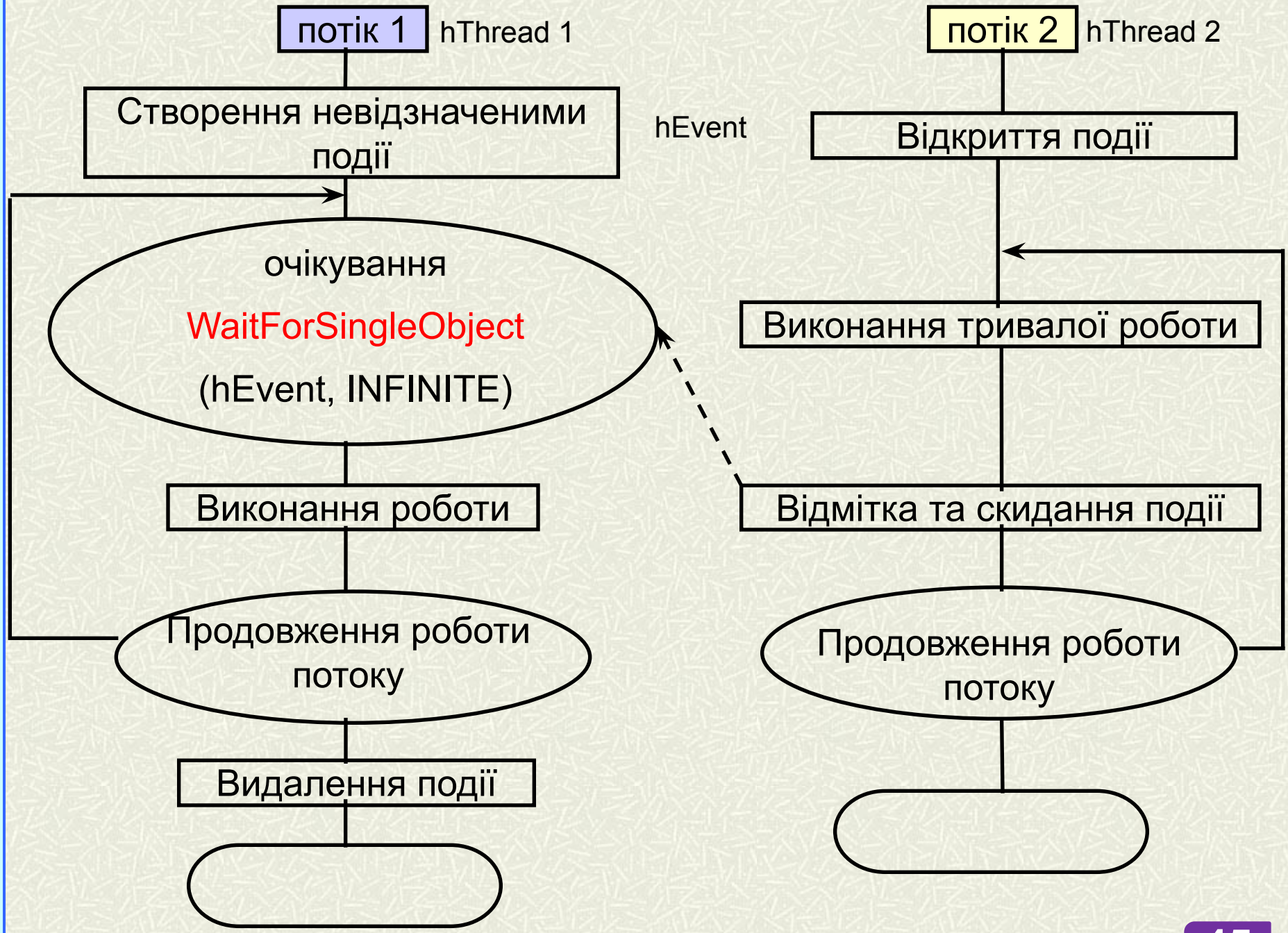


Схема використання подій

Дякую за увагу!