

# Diagramy maszyny stanowej (state machines)

---

Dr hab. inż. Ilona Bluemke

# Diagram maszyny stanowej

- Opisuje zmiany stanów obiektu na skutek zdarzeń. Do notacji używa się diagramów Davida Harel'a (1987).
- **Węzły - stany** - abstrakcja zbioru wartości atrybutów i połączeń obiektu
- **Skierowane krawędzie** - zmiany stanów etykietowane nazwami zdarzeń
- **diagram deterministyczny**

# stany

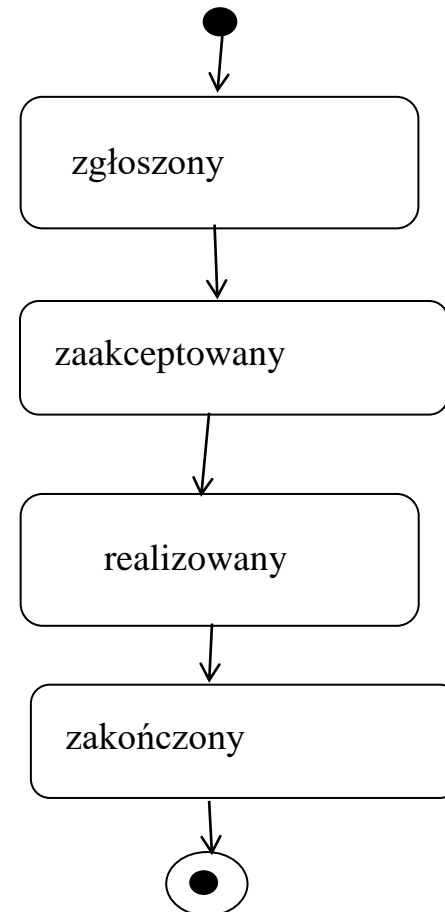
- Stan określa reakcję obiektu na zdarzenie. Reakcja może być różna dla różnych stanów. Odpowiedzią na zdarzenie może być akcja lub zmiana stanu obiektu.
- zdarzenia reprezentują chwile czasu, stany - interwały czasu.
- Stan obiektu zależy od sekwencji zdarzeń jakie obiekt otrzymał w przeszłości. Definiując stany ignorujemy atrybuty, które nie wpływają na zachowanie obiektu. Łączymy w jednym stanie wartości atrybutów, połączeń, które dają taką samą odpowiedź na zdarzenie.

# Stany charakteryzują

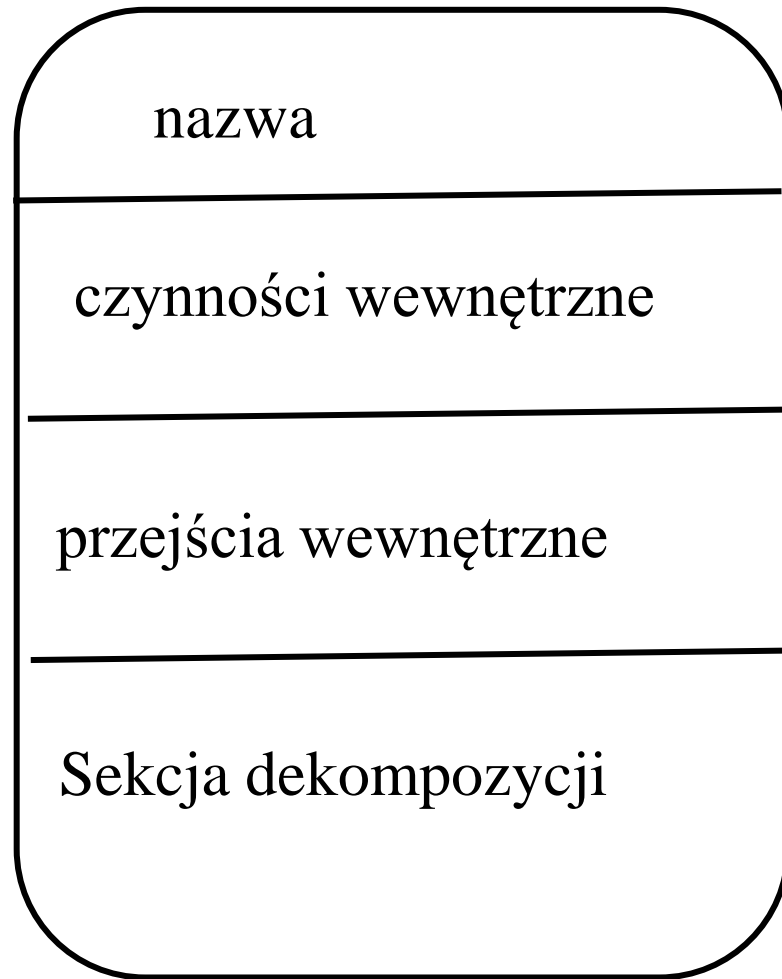
- nazwa stanu - **PRZYMIOTNIK**
- sekwencja zdarzeń powodujących wejście do tego stanu
- warunki charakteryzujące stan
- oczekiwane zdarzenia
- akcje - reakcje na zdarzenia
- stany następne
- stan początkowy ●
- stan końcowy ○

# Przykład diagramu maszyny stanowej

- Np. kurs na uczelni



# Sekcje symbolu graficznego stanu



# Diagram maszyny stanowej

- Opisuje zachowanie obiektów jednej klasy. Wszystkie instancje klasy mają takie same zachowanie - "dziela" diagram stanów. Każdy obiekt jest w swoim stanie ale jest niezależny od innych obiektów.
- Ze zdarzeniami mogą być związane akcje, zapisywane na krawędziach diagramu stanu po / , mogą także reprezentować wewnętrzne operacje sterujące np. ustawienie atrybutów, generowanie zdarzeń.

***zdarzenie/ akcja***

# Słowa kluczowe opisujące zdarzenia

- **entry** identyfikuje czynność wykonywaną przy wejściu obiektu do stanu (jedną)
- **do** identyfikuje czynność wykonywana w sposób ciągły na obiekcie znajdującym się w danym stanie, można określić kilka takich czynności (niezależnie wykonywanych), czynności te są wykonywane **po** czynności *entry*
- **exit** identyfikuje czynność wykonywaną przy wyjściu ze stanu (jedną)

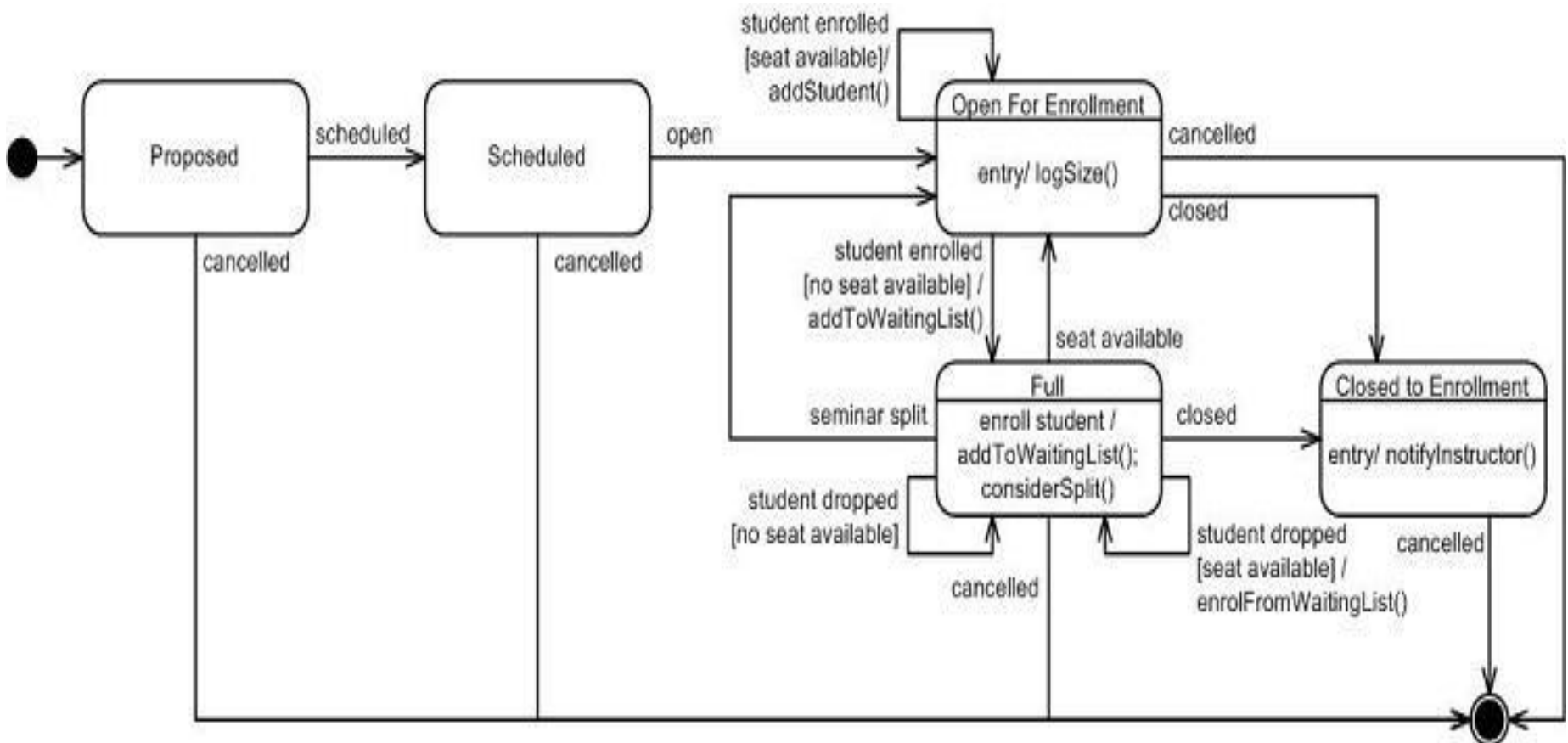


# przejścia wewnętrzne

***zdarzenie/operacja*** określają czynności wykonywane przez obiekt będący w stanie pod wpływem ***zdarzenia***

Ustaw_godz
do/ wyswielt_godz
inc/ inkrementuj_godz

# Zachowanie klasy Seminarium podczas rejestracji

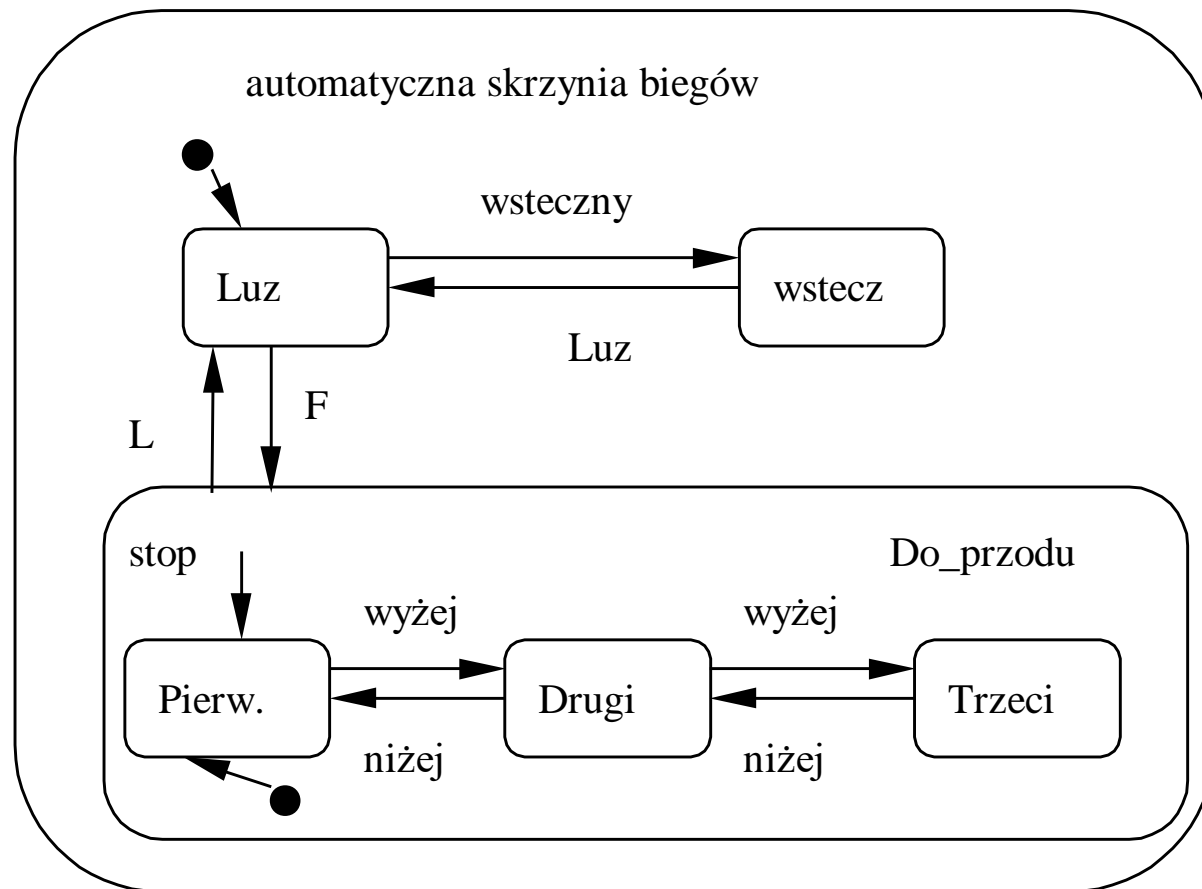


# Klasyfikacja stanów

- proste
- złożone
  - stan **złożony** zawiera maszynę stanową
  - lub jest podzielony na obszary współbieżne (zawiera podstany)

# Generalizacja stanów (relacja or)

Stan zawiera maszynę stanową



# Przykład zmian stanów

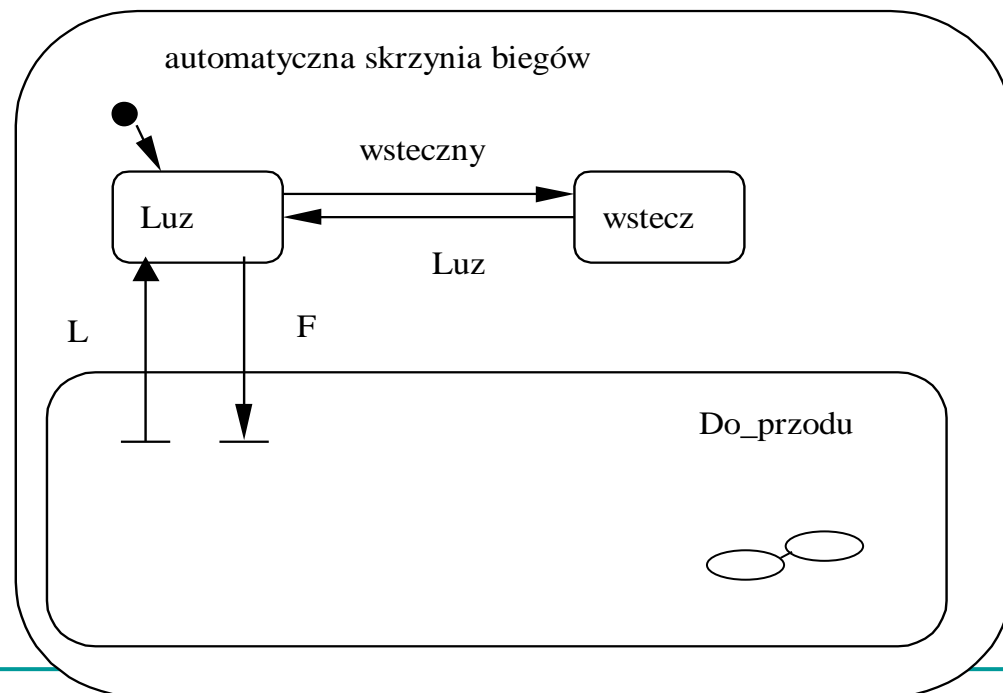
- Luz  $\rightarrow^F$  Pierw  $\rightarrow^{\text{wyżej}}$  Drugi  $\rightarrow^L$  Luz
- Luz  $\rightarrow^F$  Pierw  $\rightarrow^{\text{wyżej}}$  Drugi  $\rightarrow^{\text{wyżej}}$  Trzeci  $\rightarrow^L$  Luz

# Generalizacja stanów

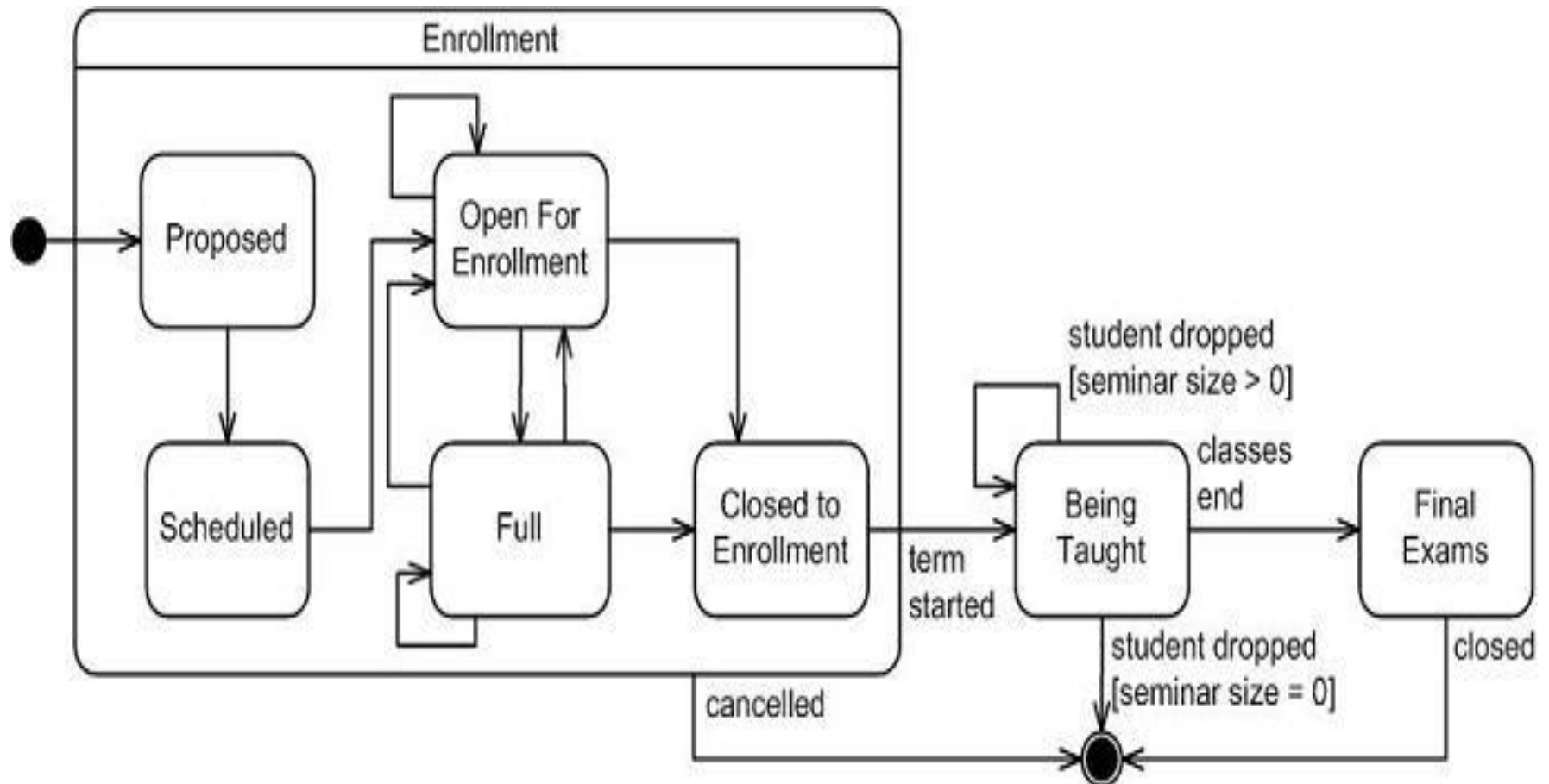
- Pozwala na opis na wysokim poziomie a następnie na uszczegóławianie na coraz niższych poziomach.
- Tworzona jest struktura hierarchiczna z dziedziczeniem wspólnego zachowania i struktury.
- Stan może mieć diagram maszyny stanowej, który dziedziczy przejścia superstanu. Przejście, akcja superstanu dotyczy wszystkich jego podstanów (chyba, że zostanie przysłonięte przez przejście w podstanie).
- Wybór "L" w dowolnym podstanie "**do przodu**" powoduje przejście do stanu "**luz**". Wybór "F" powoduje przejście do stanu "**pierwszy**". Zdarzenie "**stop**" w dowolnym podstanie "**do przodu**" powoduje przejście do stanu "**pierwszy**".

# Notacja graficzna

W złożonych diagramach stanów diagramy przejść podstawów można rysować na oddzielnych diagramach.



# Pełny diagram stanów dla klasy Seminarium



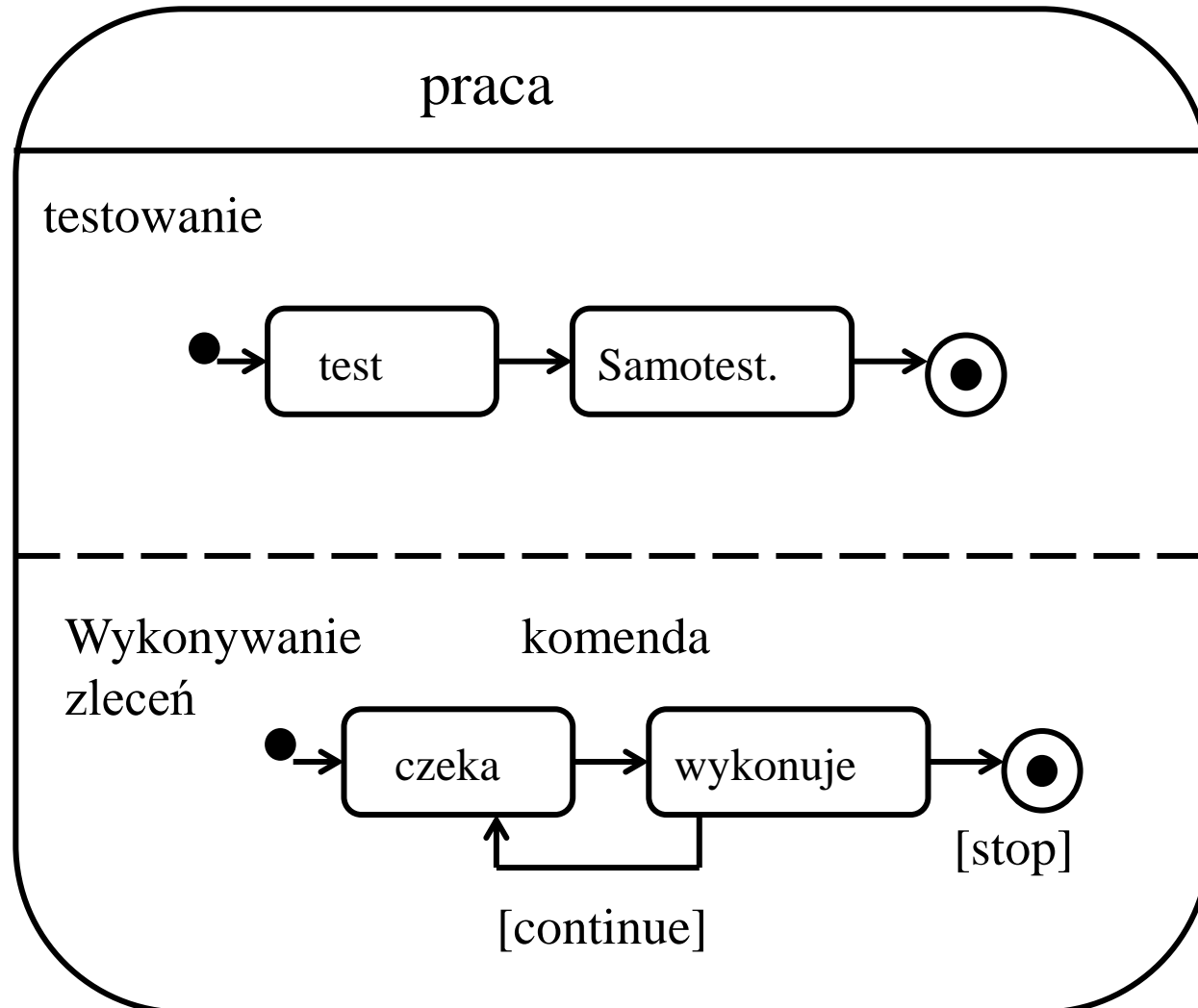


# Kompozycja – agregacja stanów

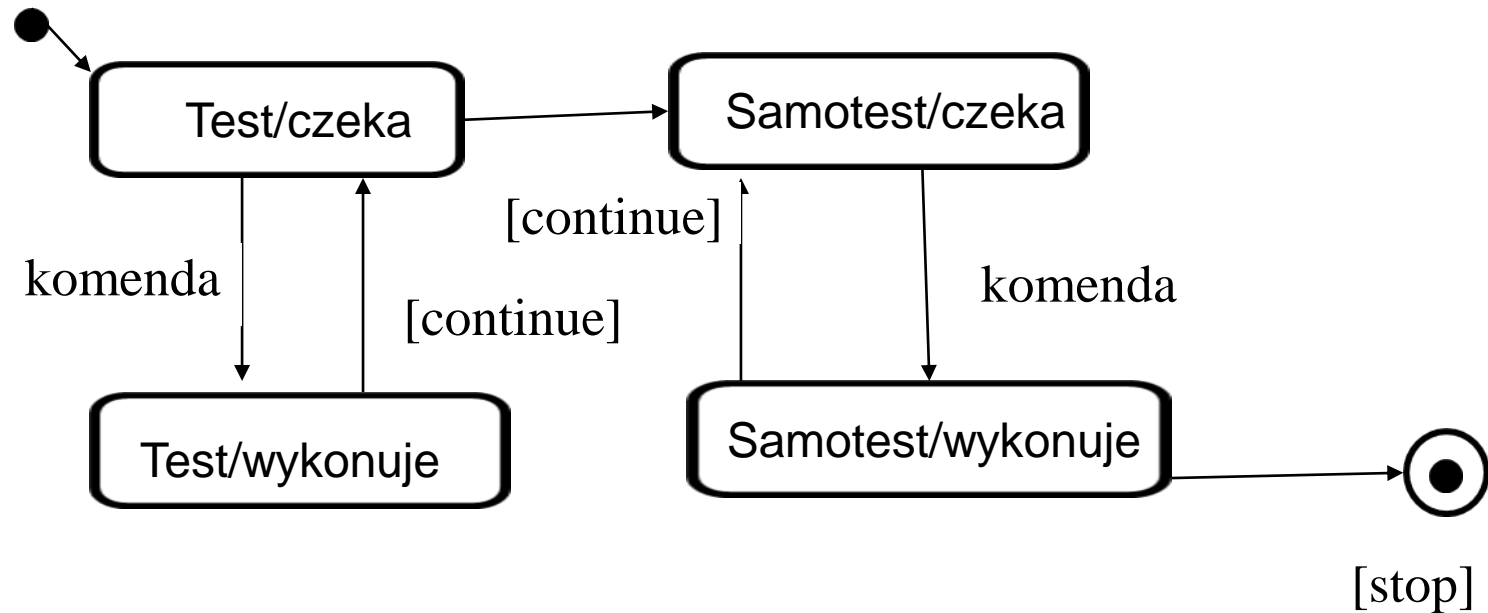
Pozwala na podział na części składowe z ograniczoną interakcją między nimi.

- Jest równoważne **współbieżności stanów**.
- Każdy komponent wykonuje przejścia równoległe z pozostałymi (równoległe działają „testowanie” i „wykonywanie zleceń” na slajdzie następnym).

# Przykład agregacji stanów



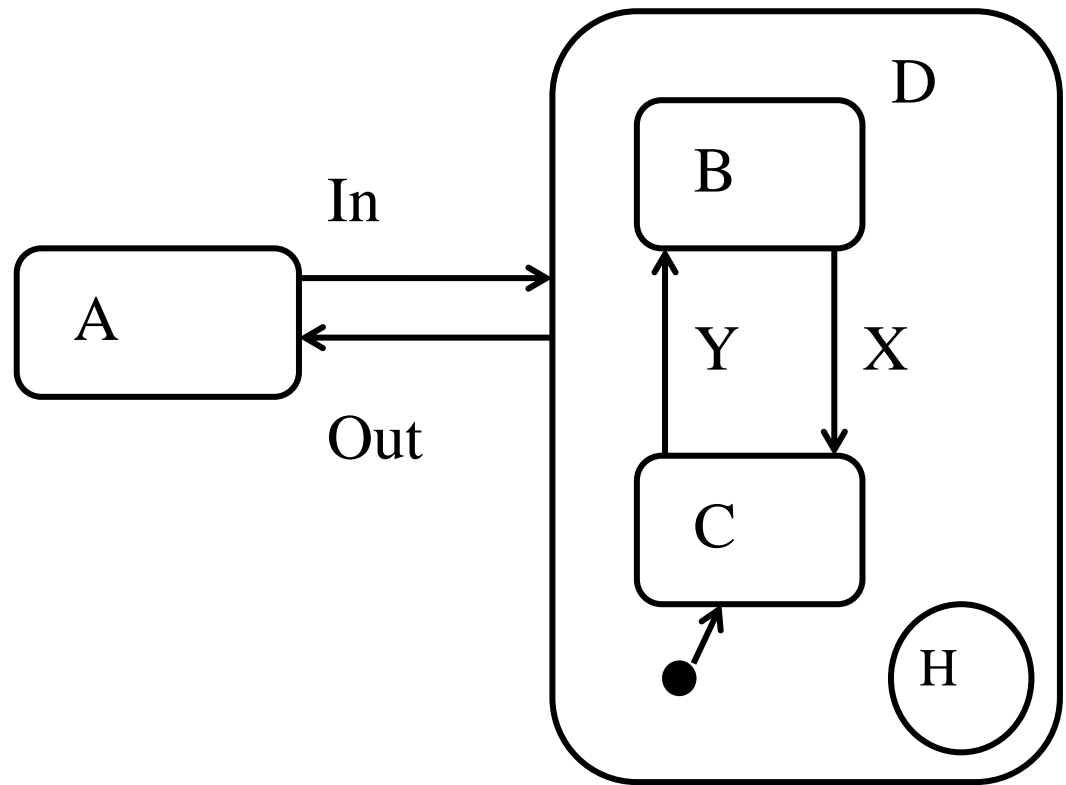
# Zadanie -1



„płaski” model - bez  
agregacji stanów

# Mechanizm historii

Pozwala na  
pamiętanie stanu  
ostatnio  
odwiedzonego w  
podstanie i  
wejście do niego  
przy kolejnym  
„wejściu” do stanu  
złożonego.

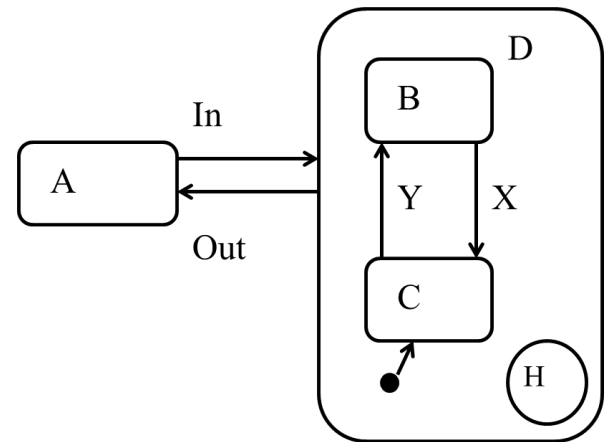


- Bez mechanizmu historii

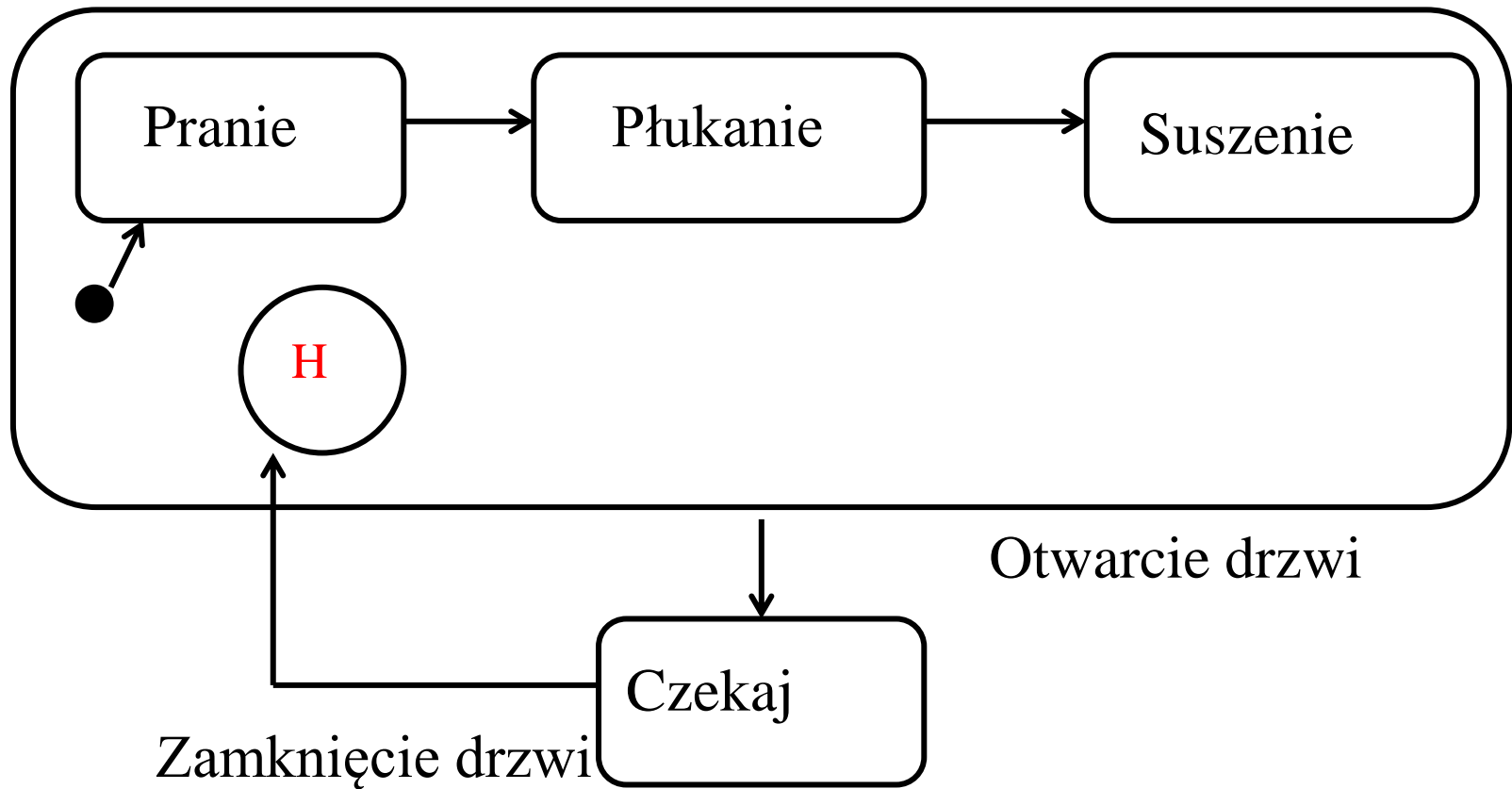
- $A \rightarrow_{(In)} C \rightarrow_{(Y)} B \rightarrow_{(Out)} A \rightarrow_{(In)} C$

- Z mechanizmem historii

- $A \rightarrow_{(In)} C \rightarrow_{(Y)} B \rightarrow_{(Out)} A \rightarrow_{(In)} B$



# Przykład maszyny stanowej z historią



# Zmiany stanów

- Bez mechanizmu historii

- Pranie -> Płukanie -> **otwarcie drzwi** -> Pranie

- Z mechanizmem historii

- Pranie -> Płukanie -> **otwarcie drzwi** -> Płukanie

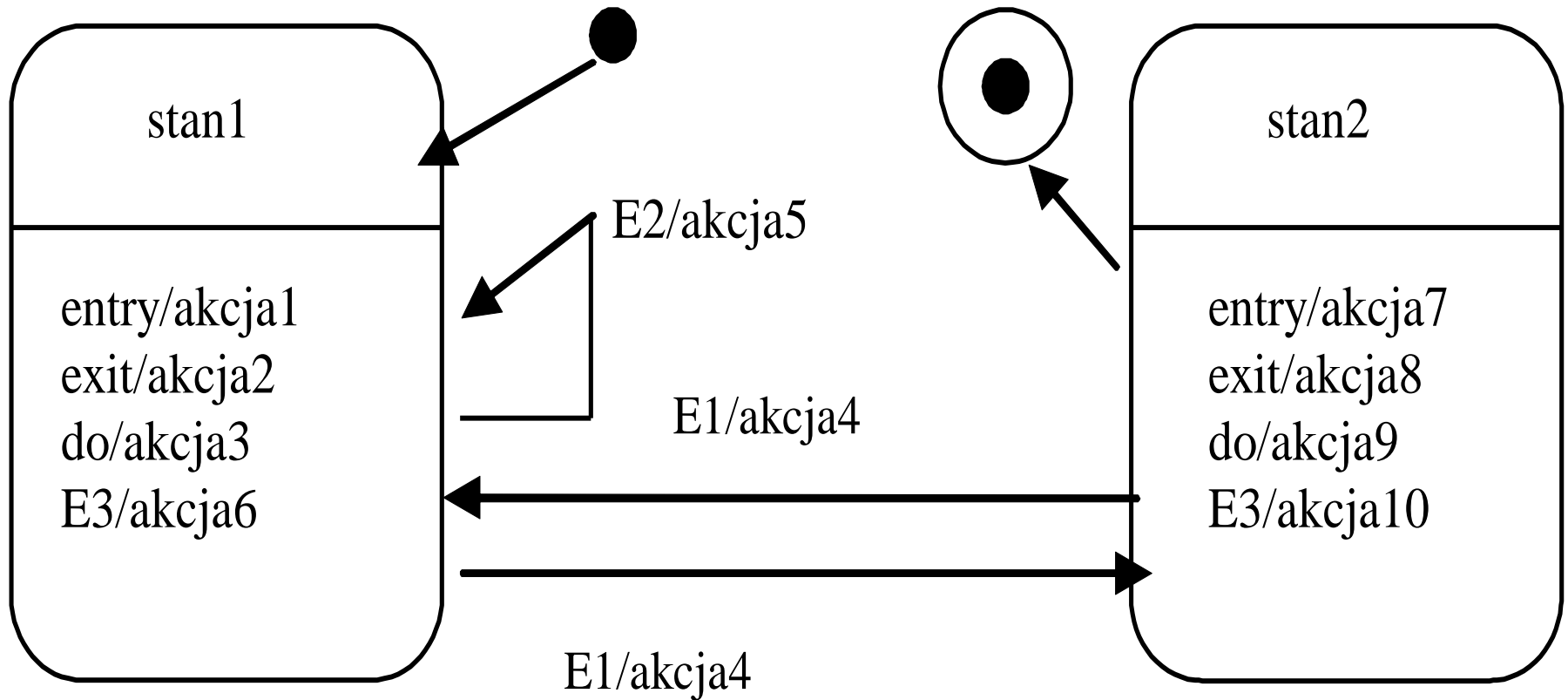
## Zadanie -2

Poniżej podano diagram zmian stanów dla pewnej klasy. Podaj jakie czynności będą kolejno wykonane przez obiekt tej klasy dla następującej sekwencji zdarzeń:

- **utworzenie obiektu,**
- **E1,**
- **E3,**
- **E2,**
- **E1.**

Uzasadnij swoje rozwiązanie.

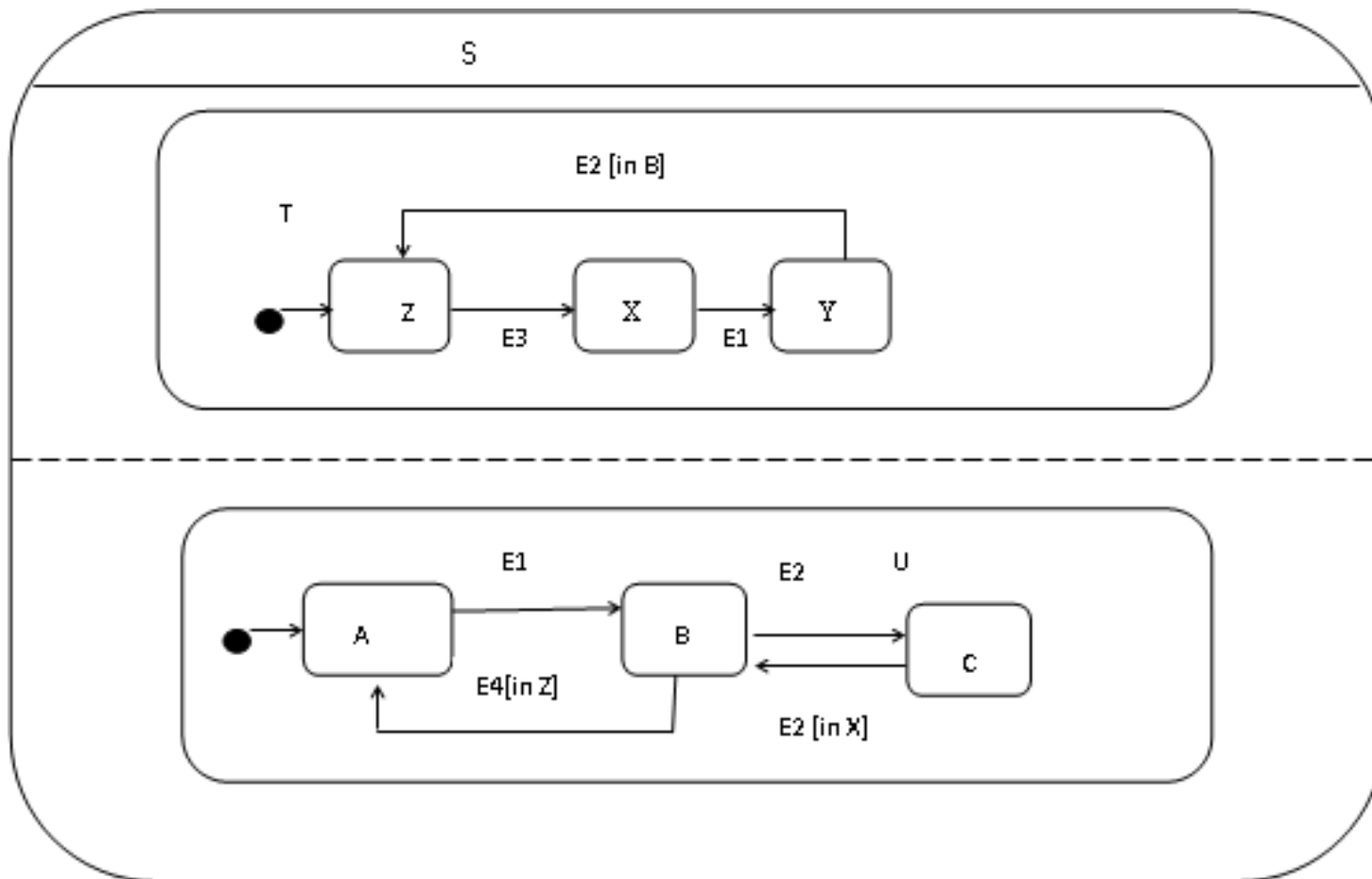




# Zadanie -1- rozwiązanie początek

zdarzenie	Stan osiągnięty	Czynności
Utworzenie obiektu	stan1	akcja1/entry; akcja3/do
E1	stan2	akcja2/exit; akcja4/E1; akcja7/entry; akcja9/do
E3	stan2	akcja10/E3; akcja9/do
E2	stan2	akcja9/do

# zadanie



# rozwiązanie

