

# 1. Foglalkozás

## Téma bevezetése:

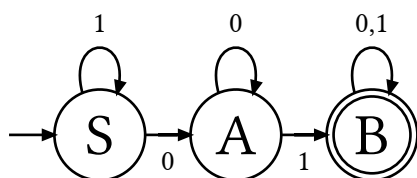
- Programozás?
- Algoritmusok?
- Állapotgépek?

DVA intuitívan - egy gép, ami betűket olvas, és észben tartja, hogy hol tart, a végén pedig eldönti, hogy a szót elfogadja, vagy sem.

## Fogalmak:

- Betű - Egy darab (szétválaszthatatlan) karakter.
- Ábécé - betűk halmaza. ( $\Sigma$ )
- Szó - Betűk egymásutánja.
- Nyelv - Szavak halmaza.
- $a^n$   $a$  betű  $n$ -szer egymás után leírva, ahol  $a \in \Sigma, n \in \mathbb{N}$

## Jelölések:



- Állapot - Nagy (latin) betű.
- Átmenet - Állapotok közti nyíl
- Elfogadó állapot - dula karika
- $\varepsilon$  - üres szó
- minden állapotból kell átmenet az ábécé minden betűjére
- minden állapotból minden betű csak egyszer
- **szóvá kell tenni, hogy ez a DVA, lesz más is, ahol kicsit mások a szabályok**

## Közös feladatmegoldás

## Önálló feladatmegoldás

# 2. Foglalkozás

## Hiányos DVA

- elhagyhatók átmenetek
- nem definiált átmenet esetén azonnal elutasít
- **feladatokon keresztül vegyék észre, hogy ez is egy DVA-vá átalakítható**

## Feladatok

### NVA

- egy állapotban egy betűre több átmenet definiálható
- ha van egy megfelelő lefutás, akkor a szót elfogadjuk
- általában hiányos is (hogy kevesebbet kelljen írni)

## Feladatok

## NVA és DVA ekvivalens

- szerintük melyik az “erősebb?”
- mutassuk meg, hogy NVA alakítható DVA-vá
- vegyük észre, hogy a DVA egy NVA

### 3. Foglalkozás

#### “Verseny”

(Ezt nem én találtam ki, de a tematika tanításakor népszerű szokott lenni)

- Gyerekek rendeződjenek 4-5 fős csapatokba
- Minden csapat találjon ki egy automatát annyi állapottal, ahányan vannak
- Szimulálják az automatát
- A többi csapat adjon inputokat, és próbálja meg kitalálni, hogy milyen automatát szimulálnak

### 4. Foglalkozás

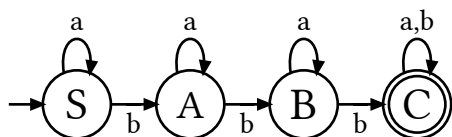
#### Nem mindent lehet NVA-val kifejezni

- példanyelv, amire nem tudnak automatát adni
- veremautomaták bevezetése
- közös feladatmegoldás

## Feladatsor megoldásokkal

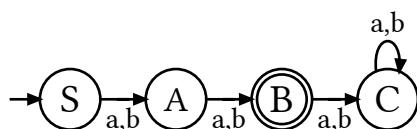
Ahol a feladat mást nem mond, az ábécé legyen  $\Sigma = \{a, b\}$ .

- Adj meg egy determinisztikus véges automatát, mely azokat a szavakat fogadja el, amelyekben szerepel legalább 3 darab  $a$  betű.

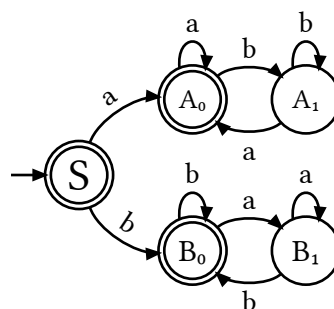


- Adj determinisztikus véges automatát a következő nyelvekre:

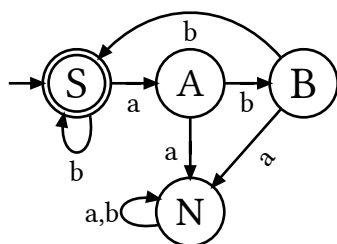
- 2 betűből álló szavak



- szavak, melyek első és utolsó betűje megegyezik



- szavak, melyekben minden "a" után "bb" következik



- szavak, melyekben a "aa" részszo pontosan egyszer szerepel

