

Záróvizsga tételsor mérnökinformatikus hallgatóknak*

Palkovics Dénes,

2019

A záróvizsga tematikája és tartalma

A záróvizsgán kettő kérdésre kell válaszolni, egyre az általános kérdések közül, egyre pedig a specializációnak megfelelő kérdések közül

Tartalomjegyzék

1. Általános kérdések	5
1.1. Az informatika logikai alapjai	5
1.1.1. Az elsőrendű matematikai logikai nyelv.	5
1.1.2. A nyelv interpretációja, formulák igazságértéke az interpretációban adott változókiértékelés mellett.	5
1.1.3. Logikai törvény, logikai következmény.	7
1.1.4. Logikai ekvivalencia, normálformák.	7
1.1.5. Kalkulusok (Gentzen-kalkulus).	7
1.2. Operációs rendszerek	10
1.2.1. Operációs rendszerek fogalma, felépítése, osztályozásuk.	10
1.2.2. Az operációs rendszerek jellemzése (komponensei és funkciói).	10
1.2.3. A rendszeradminisztráció, fejlesztői és alkalmazói támogatás eszközei.	12
1.3. Magas szintű Programozási nyelvek	13
1.3.1. Adattípus, konstans, változó, kifejezés.	13
1.3.2. Paraméterkiértékelés, paraméterátadás.	13
1.3.3. Hatáskör, névterek, élettartam.	13
1.3.4. Fordítási egységek, kivételkezelés.	13
1.4. Magas Sintű programozási nyelvek 2	14
1.4.1. Speciális programnyelvi eszközök.	14
1.4.2. Az objektumorientált programozás eszközei és jelentősége.	14
1.4.3. Funkcionális és logikai programozás.	14
1.5. Adatszerkezetek és algoritmusok	15
1.5.1. Adatszerkezetek reprezentációja.	15
1.5.2. Műveletek adatszerkezetekkel.	15
1.5.3. Adatszerkezetek osztályozása és jellemzésük.	15
1.5.4. Szekvenciális adatszerkezetek: sor, verem, lista, sztring.	15
1.5.5. Egyszerű és összetett állományszerkezetek.	15
1.6. Adatbázisrendszerek	16

*A Debreceni Egyetem mérnökinformatikus alapszakhoz

1.6.1.	Relációs, ER és objektumorientált modellek jellemzése.	16
1.6.2.	Adatbázisrendszer.	16
1.6.3.	Funkcionális függés.	16
1.6.4.	Relációalgebra és relációkalkulus.	16
1.6.5.	Az SQL.	16
1.7.	Hálózati architektúrák	17
1.7.1.	Az ISO OSI hivatkozási modell.	17
1.7.2.	Ethernet szabványok.	17
1.7.3.	A hálózati réteg forgalomirányító mechanizmusai.	17
1.7.4.	Az internet hálózati protokollok, legfontosabb szabványok és szolgáltatások.	17
1.8.	Fizika 1	18
1.8.1.	Fizikai fogalmak, mennyiségek.	18
1.8.2.	Impulzus, impulzusmomentum.	18
1.8.3.	Newton törvényei.	18
1.8.4.	Munkatétel.	18
1.8.5.	Az I. és II. főtétele.	18
1.8.6.	A kinetikus gázmodell.	18
1.9.	Fizika 2	19
1.9.1.	Elektromos alapfogalmak és alapjelenségek.	19
1.9.2.	Ohm-törvény.	19
1.9.3.	A mágneses tér tulajdonságai.	19
1.9.4.	Elektromágneses hullámok.	19
1.9.5.	A Bohr-féle atommodell.	19
1.9.6.	A radioaktív sugárzás alapvető tulajdonságai.	19
1.10.	Elektronika 1, 2	20
1.10.1.	Passzív áramköri elemek tulajdonságai, RC és RLC hálózatok.	20
1.10.2.	Diszkrét félvezető eszközök, aktív áramköri elemek, alapkapcsolások.	20
1.10.3.	Integrált műveleti erősítők.	20
1.10.4.	Tápegységek.	20
1.10.5.	Mérőműszerek.	20
1.11.	Digitális Technika	21
1.11.1.	Logikai függvények kapcsolástechnikai megvalósítása.	21
1.11.2.	Digitális áramköri családok jellemzői(TTL, CMOS, NMOS).	21
1.11.3.	Különböző áramköri családok csatlakoztatása.	21
1.11.4.	Kombinációs és szekvenciális hálózatok. A/D és D/A átalakítók.	21

2. Infokommunikációs hálózatok	22
2.1. Távközlő hálózatok	22
2.1.1. Fizikai jelátviteli közegek.	22
2.1.2. Forráskódolás, csatornakódolás és moduláció.	22
2.1.3. Csatornafelosztás és multiplexelési technikák.	22
2.1.4. Vezetékes és a mobil távközlő hálózatok.	22
2.1.5. Műholdas kommunikáció és helymeghatározás.	22
2.2. Hálózatok hatékonyságanalízise	23
2.2.1. Markov-láncok, születési-kihalási folyamatok.	23
2.2.2. A legalapvetőbb sorbanállási rendszerek vizsgálata.	23
2.2.3. A rendszerjellemzők meghatározásának módszerei, meghatározásuk számítógépes támogatása. .	23
2.3. Adatbiztonság	24
2.3.1. Fizikai, ügyviteli és algoritmusos adatvédelem, az informatikai biztonság szabályozása.	24
2.3.2. Kriptográfiai alapfogalmak.	24
2.3.3. Klasszikus titkosító módszerek.	24
2.3.4. Digitális aláírás, a DSA protokoll.	24
2.4. A RIP protokoll működése és paramétereinek beállítása (konfigurációja).	25
2.5. Bevezetés a Cisco eszközök programozásába 1	26
2.5.1. A forgalomszűrés, forgalomszabályozás (Trafficfiltering, ACL) céljai és beállítása (konfigurációja) egy választott példa alapján.	26
2.6. Bevezetés a Cisco eszközök programozásába 2	27
2.6.1. A forgalomirányítási táblázatok felépítése, statikus és dinamikus routing összehasonlítása. . . .	27
Tárgymutató	28

1. Általános kérdések

1.1. Az informatika logikai alapjai

1.1.1. Az elsőrendű matematikai logikai nyelv.

Definíció (Elsőrendű nyelv). *Klasszikus elsőrendű nyelven az*

$$L^{(1)} = \langle LC, Var, Con, Term, Form \rangle$$

rendezett ötöst értjük, ahol

1. $LC = \{\neg, \supset, \wedge, \vee, \equiv, =, \forall, \exists, (,)\}$ a nyelv logikai konstansainak halmaza¹
2. $Var = \{x_n | n = 0, 1, 2, \dots\}$ a nyelv változóinak megszámlálhatóan végtelen halmaza²
3. $Con = \bigcup_{n=0}^{\infty} (\mathcal{F}(n) \cup \mathcal{P}(n))$ a nyelv nemlogikai konstansainak legfeljebb megszámlálhatóan végtelen halmaza³
 - a) $\mathcal{F}(0)$ a névparaméterek (névkonstansok),
 - b) $\mathcal{F}(n)$ az n argumentumú függvényjelek (műveleti jelek),
 - c) $\mathcal{P}(0)$ a állításparaméterek (állításkonstansok),
 - d) $\mathcal{P}(n)$ az n argumentumú predikátumparaméterek (predikátumkonstansok) halmaza.
4. Az $LC, Var, \mathcal{F}(n), \mathcal{P}(n)$ halmazok ($n = 0, 1, 2, \dots$) páronként diszjunktak.
5. A nyelv terminusainak a halmazát, azaz a $Term$ halmazt az alábbi induktív definíció adja:
 - a) $Var \cup \mathcal{F}(0) \subseteq Term$
 - b) Ha $f \in \mathcal{F}(n)$, ($n = 1, 2, \dots$), és $t_1, t_2, \dots, t_n \in Term$, akkor $f(t_1, t_2, \dots, t_n) \in Term$
6. A nyelv formuláinak halmazát, azaz a $Form$ halmazt az alábbi induktív definíció adja meg:
 - a) $\mathcal{P} \subseteq Form$
 - b) Ha $t_1, t_2 \in Term$, akkor $(t_1 = t_2) \in Form$
 - c) Ha $P \in \mathcal{P}$, ($n = 1, 2, \dots$), és $t_1, t_2, \dots, t_n \in Term$, akkor $P(t_1, t_2, \dots, t_n) \in Form$
 - d) Ha $A \in Form$, akkor $\neg A \in Form$
 - e) Ha $A, B \in Form$, akkor $(A \supset B), (A \wedge B), (A \vee B), (A \equiv B) \in Form$
 - f) Ha $x \in Var, A \in Form$, akkor $\forall x A, \exists x A \in Form$

Megjegyzés. Azokat a formulákat, amelyek a 6. a), b), c) szabályok által jönnek létre, atomi formuláknak vagy prímmuláknak nevezzük.

1.1.2. A nyelv interpretációja, formulák igazságértéke az interpretációban adott változókiértékelés mellett.

Definíció (interpretáció (elsőrendű)). Az $\langle U, \rho \rangle$ párt az $L^{(1)}$ nyelv egy interpretációjának nevezzük, ha

1. $U \neq \emptyset$ azaz U nemüres halmaz
2. $Dom(\rho) = Con$ azaz a ρ a Con halmazon értelmezett függvény, amelyre teljesülnek a következők:

¹ A logikai konstansok olyan nyelvi eszközök, amelyek jelentését a szemantikai szabályok (logikai kalkulusok esetén az axiómák) rögzítik. Egy adott logikai rendszer esetén a logikai konstansok rögzített jelentéssel (rögzített szemantikai értékkel) rendelkeznek, jelentésük (szemantikai értékük) minden interpretációban megegyezik. Egy adott logikai rendszer esetén a logikai konstansokat általában az adott logikai rendszer nyelvének LC halmaza tartalmazza.

² A köznyelvi mondatokban nevek helyett néha névmásokkal utalunk egyes individuumokra (objektumokra). A tudományos nyelvben gyakran kívánatos analóg kifejezési formák megadása. A szabadság, az egyértelműség és a tömörség érdekében ilyenkor mesterséges névmásokat vezetnek be, amelyeket változóknak neveznek.

³ A nemlogikai konstansok, más néven paraméterek olyan nyelvi eszközök, amelyek jelentését az interpretáció rögzíti. Egy adott logikai rendszer esetén a nemlogikai konstansok (a paraméterek) nem rendelkeznek rögzített jelentéssel (rögzített szemantikai értékkel), jelentésük (szemantikai értékük) interpretációról interpretációra változhat. Egy adott logikai rendszer esetén a nemlogikai konstansokat általában az adott logikai rendszer nyelvének Con halmaza tartalmazza.

a) Ha $a \in F(0)$, akkor $\rho(a) \in U$

b) Ha $f \in \mathcal{F}(n)$ ahol $n \neq 0$, akkor $\rho(f)$ az $U^{(n)}$ halmazon értelmezett az U halmazba képező függvény ($\rho(f) : U^{(n)} \rightarrow U$)

c) Ha $p \in \mathcal{P}(0)$, akkor $\rho(p) \in 0, 1$

d) Ha $P \in \mathcal{P}(n)$ ahol $n \neq 0$, akkor $\rho(P) \subseteq U^{(n)}$

Definíció (értékelés (elsőrendű)). Legyen $L^{(1)} = \langle LC, Var, Con, Term, Form \rangle$ egy elsőrendű nyelv, $\langle U, \rho \rangle$ pedig a nyelv egy interpretációja. Az $\langle U, \rho \rangle$ interpretációra támaszkodó ν értékelésen egy olyan függvényt értünk, amely teljesíti a következőket:

- $Dom(\nu) = Var$
- Ha $x \in Var$, akkor $\nu(x) \in U$

Definíció (értékelés (elsőrendű)). Legyen $L^{(1)} = \langle LC, Var, Con, Term, Form \rangle$ egy elsőrendű nyelv, $\langle U, \rho \rangle$ pedig a nyelv egy interpretációja, ν pedig az $\langle U, \rho \rangle$ interpretációra támaszkodó értékelés.

1. Ha $a \in F(0)$, akkor $|a|_{\nu}^{\langle U, \rho \rangle} = \rho(a)$

2. Ha $x \in Var$, akkor $|x|_{\nu}^{\langle U, \rho \rangle} = \nu(x)$

3. Ha $f \in F(n)$, ($n = 1, 2, \dots$) és $t_1, t_2, \dots, t_n \in Term$, akkor

$$|f(t_1, t_2, \dots, t_n)|_{\nu}^{\langle U, \rho \rangle} = \rho(f)(|t_1|_{\nu}^{\langle U, \rho \rangle}, |t_2|_{\nu}^{\langle U, \rho \rangle}, \dots, |t_n|_{\nu}^{\langle U, \rho \rangle})$$

4. Ha $p \in P(0)$, akkor $|p|_{\nu}^{\langle U, \rho \rangle} = \rho(p)$

5. Ha $t_1, t_2 \in Term$, akkor

$$|(t_1 = t_2)|_{\nu}^{\langle U, \rho \rangle} = \begin{cases} 1, & \text{ha } |t_1|_{\nu}^{\langle U, \rho \rangle} = |t_2|_{\nu}^{\langle U, \rho \rangle} \\ 0, & \text{egyébként.} \end{cases} \quad (1)$$

6. Ha $P \in P(n)$ ahol $n = 0, 1, \dots, n \in Term$, akkor

$$|P(t_1, t_2, \dots, t_n)|_{\nu}^{\langle U, \rho \rangle} = \begin{cases} 1, & \text{ha } (|t_1|_{\nu}^{\langle U, \rho \rangle}, |t_2|_{\nu}^{\langle U, \rho \rangle}, \dots, |t_n|_{\nu}^{\langle U, \rho \rangle}) \in \rho(P) \\ 0, & \text{egyébként.} \end{cases} \quad (2)$$

7. Ha $A \in Form$, akkor $|\neg A|_{\nu}^{\langle U, \rho \rangle} = 1 - |A|_{\nu}^{\langle U, \rho \rangle}$.

8. Ha $A, B \in Form$, akkor

$$|(A \supset B)|_{\nu}^{\langle U, \rho \rangle} = \begin{cases} 0, & \text{ha } |A|_{\nu}^{\langle U, \rho \rangle} = 1, \text{ és } |B|_{\nu}^{\langle U, \rho \rangle} = 0 \\ 1, & \text{egyébként.} \end{cases} \quad (3)$$

$$|(A \wedge B)|_{\nu}^{\langle U, \rho \rangle} = \begin{cases} 1, & \text{ha } |A|_{\nu}^{\langle U, \rho \rangle} = 1, \text{ és } |B|_{\nu}^{\langle U, \rho \rangle} = 1 \\ 0, & \text{egyébként.} \end{cases} \quad (4)$$

$$|(A \vee B)|_{\nu}^{\langle U, \rho \rangle} = \begin{cases} 0, & \text{ha } |A|_{\nu}^{\langle U, \rho \rangle} = 0, \text{ és } |B|_{\nu}^{\langle U, \rho \rangle} = 0 \\ 1, & \text{egyébként.} \end{cases} \quad (5)$$

$$|(A \equiv B)|_{\nu}^{\langle U, \rho \rangle} = \begin{cases} 1, & \text{ha } |A|_{\nu}^{\langle U, \rho \rangle} = |B|_{\nu}^{\langle U, \rho \rangle} \\ 0, & \text{egyébként.} \end{cases} \quad (6)$$

9. Ha $A \in Form$, $x \in Var$, akkor

$$|(\forall x A)|_{\nu}^{\langle U, \rho \rangle} = \begin{cases} 0, & \text{ha van olyan } u \in U, \text{ hogy } |A|_{\nu[x:u]}^{\langle U, \rho \rangle} = 0 \\ 1, & \text{egyébként.} \end{cases} \quad (7)$$

$$|(\exists x A)|_{\nu}^{\langle U, \rho \rangle} = \begin{cases} 1, & \text{ha van olyan } u \in U, \text{ hogy } |A|_{\nu[x:u]}^{\langle U, \rho \rangle} = 1 \\ 0, & \text{egyébként.} \end{cases} \quad (8)$$

1.1.3. Logikai törvény, logikai következmény.

Definíció (modell). Legyen $L^{(1)} = (LC, Var, Con, Term, Form)$ egy elsőrendű nyelv és $\Gamma \subseteq Form$ egy tetszőleges formulahalmaz. Az (U, ρ, ν) rendezett hármas elsőrendű modellje a Γ formulahalmaznak, ha

- (U, ρ) egy interpretációja az $L^{(1)}$ nyelvnek;
- ν egy (U, ρ) interpretációra támaszkodó értékelés;
- minden $A \in \Gamma$ esetén $|A|_{\nu}^{(U, \rho)} = 1$.

Definíció. Legyen $L^{(1)} = (LC, Var, Con, Term, Form)$ egy elsőrendű nyelv és $\Gamma \subseteq Form$ egy tetszőleges formulahalmaz, $A, B \in Form$ egy tetszőleges formulák.

- Egy Γ formulahalmaz kielégíthető, ha van (elsőrendű) modellje;
- Egy Γ formulahalmaz kielégíthetetlen, ha nem kielégíthető, azaz nincs modellje;
- Az A formula modellje az $\{A\}$ egyelemű formulahalmaz modelljét értjük;
- Az A formula kielégíthető, ha $\{A\}$ formulahalmaz kielégíthető;
- Az A formula kielégíthetetlen, ha $\{A\}$ formulahalmaz kielégíthetetlen;
- A Γ formulahalmaznak logikai következménye az A formula, ha a $\Gamma \cup \{\neq A\}$ formulahalmaz kielégíthetetlen. Jelölés: $\Gamma \models A$
- Az A formulának logikai következménye a B formula, ha a $\{A\} \models B$. Jelölés: $A \models B$
- Az A formula érvényes (logikai törvény), ha $\emptyset \models A$, azaz ha az A formula logikai következménye az üres halmaznak. Másképpen, ha minden (U, ρ) interpretációjában, minden ν értékelés szempontjából $|A|_{\nu}^{(U, \rho)} = 1$ Jelölés: $\models A$
- Az A és a B formula logikailag ekvivalens, ha $A \models B$ és $B \models A$. Jelölés: $A \Leftrightarrow B$

1.1.4. Logikai ekvivalencia, normálformák.

Definíció (Logikai ekvivalencia). lásd:a 1.1.3 fejezet definíciója.

Definíció (elemi konjunkció). Legyen $L^{(0)} = (LC, Con, Form)$ egy nulladrendű nyelv. Ha az $A \in Form$ formula literál vagy különböző alapú literálok konjunkciója, akkor A -t elemi konjunkciónak nevezzük.

Definíció (elemi diszjunkció). Legyen $L^{(0)} = (LC, Con, Form)$ egy nulladrendű nyelv. Ha az $A \in Form$ formula literál vagy különböző alapú literálok diszjunkciója, akkor A -t elemi diszjunkciónak nevezzük.

Definíció (diszjunktív normálforma). Egy elemi konjunkciót vagy elemi konjunkciók diszjunkcióját diszjunktív normálformának nevezzük.

Definíció (konjunktív normálforma). Egy elemi diszjunktív vagy elemi diszjunktív konjunkcióját konjunktív normálformának nevezzük.

Definíció. Legyen $L^{(0)} = (LC, Con, Form)$ egy nulladrendű nyelv és $A \in Form$ egy formula. Ekkor létezik olyan $B \in Form$, hogy

- $A \Leftrightarrow B$
- B diszjunktív vagy konjunktív normálformájú.

1.1.5. Kalkulusok (Gentzen-kalkulus).

Logikai kalkulus Logikai kalkuluson olyan adott nyelv formuláihoz tartozó formális rendszert, szabályrendszert értünk, amely pusztán szintaktikailag, szemantika nélkül ad meg egy következményrelációt. A logikai kalkulus tehát egy axiómarendszer, amely magában a logikai tautológiákat állítja elő, adott formulákat ideiglenesen hozzávéve (premissza) pedig más formulákra (konklúzió) lehet jutni (következtetni) vele.

Gentzen-féle szekvenciakalkulus Ebben a kalkulusban nem formulákra vonatkoznak a szabályok és nem is formulák alkotják az axiómákat, hanem a formulák eddigi szerepét az ún. szekvencia töltik be. Szekvenciának nevezzük a

$$\Gamma \vdash \Delta$$

alakú jelsorozatot, ahol Γ és Δ olyan rendezett jelsorozatok, amelyeknek minden tagja egy formula.

Definíció (axiómasémák). Legyen $L^{(0)} = (LC, Con, Form)$ egy nulladrendű nyelv (a klasszikus állításlogika nyelve). A nulladrendű kalkulus (klasszikus állításkalkulus) axiómasémái (alapsémái):

1. $A \supset (B \supset A)$
2. $(A \supset (B \supset C)) \supset ((A \supset B) \supset (A \supset C))$
3. $(\neg A \supset \neg B) \supset (B \supset A)$

Az axiómaséma szabályos behelyettesítésén olyan formulát értünk, amely az axiómasémából a benne szereplő betűk tetszőleges formulával való helyettesítése útján jön létre. A nulladrendű kalkulus (klasszikus állításkalkulus) axiómái az axiómasémák szabályos behelyettesítései.

Definíció (szintaktikai következmény). Legyen $L^{(0)} = (LC, Con, Form)$ egy nulladrendű nyelv, $\Gamma \subseteq Form$ egy tetszőleges formulahalmaz. A Γ formulahalmaz szintaktikai következményeinek induktív definíciója:

Bázis:

- Ha $A \in \Gamma$, akkor $\Gamma \vdash A$
- Ha A axióma, akkor $\Gamma \vdash A$.

Szabály (leválasztási szabály):

- Ha $\Gamma \vdash B$, és $\Gamma \vdash (B \supset A)$, akkor $\Gamma \vdash A$.

Definíció (szintaktikai következmény). Legyen $L^{(0)} = (LC, Con, Form)$ egy nulladrendű nyelv és $A, B \in Form$ két tetszőleges formula. Az A formulának szintaktikai következménye a B formula, ha $\{A\} \vdash B$. Jelölés: $A \vdash B$

Definíció (szekvencia). Legyen $L^{(0)} = (LC, Con, Form)$ egy nulladrendű nyelv, $\Gamma \subseteq Form$ egy formulahalmaz és $A \in Form$ egy formula. Ha az A formula szintaktikai következménye a Γ formulahalmaznak, akkor a $\Gamma \vdash A$ jelsorozatot szekvenciának nevezzük.

Definíció (levezethetőség). Legyen $L^{(0)} = (LC, Con, Form)$ egy nulladrendű nyelv és $A \in Form$ egy tetszőleges formula. Az A formula levezethető, ha $\emptyset \vdash A$, azaz ha az A formula szintaktikai következménye az üres halmaznak. Jelölés: $\vdash A$

Definíció (természetes levezetés szabályai). Legyen $L^{(0)} = (LC, Con, Form)$ egy nulladrendű nyelv $\Gamma, \Delta \subseteq Form$ és $A, B, C \in Form$. A természetes levezetés által az $L^{(0)}$ nyelvben bizonyítható következményrelációk alábbiak:

Bázis:

$$\frac{\omega}{\Gamma, A \vdash A} \quad (9)$$

Szabályok:

- **Struktúrális szabályok:**

- Bővítés $\frac{\Gamma \vdash A}{\Gamma, B \vdash A}$
- Felcserélés $\frac{\Gamma, B, C, \Delta \vdash A}{\Gamma, C, B, \Delta \vdash A}$
- Szűkítés $\frac{\Gamma, B, B, \Delta \vdash A}{\Gamma, B, \Delta \vdash A}$
- Metszet $\frac{\Gamma \vdash A, \Delta, A \vdash B}{\Gamma, \Delta \vdash B}$

- **Logikai szabályok:**

- Implikáció szabályai:
 - * bevezető: $\frac{\Gamma, A \vdash B}{\Gamma \vdash A \supset B}$
 - * alkalmazó: $\frac{\Gamma \vdash A, \Gamma \vdash A \supset B}{\Gamma \vdash B}$
- Negáció szabályai:

- * *bevezető*: $\frac{\Gamma, A \vdash B \quad \Gamma, A \vdash \neg B}{\Gamma \vdash \neg A}$
- * *alkalmazó*: $\frac{\Gamma \vdash \neg \neg A}{\Gamma \vdash A}$
- *Konjunkció szabályai*:
 - * *bevezető*: $\frac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}$
 - * *alkalmazó*: $\frac{\Gamma, A, B \vdash C}{\Gamma, A \wedge B \vdash C}$
- *Diszjunkció szabályai*:
 - * *bevezető*: $\frac{\Gamma \vdash A}{\frac{\Gamma \vdash A \vee B \quad \Gamma \vdash B}{\Gamma \vdash A \vee B}}$
 - * *alkalmazó*: $\frac{\Gamma, A \vdash C \quad \Gamma, B \vdash C}{\Gamma, A \vee B \vdash C}$
- *(Materiális) ekvivalencia szabályai*:
 - * *bevezető*: $\frac{\Gamma, A \vdash B \quad \Gamma, B \vdash A}{\Gamma \vdash A \equiv B}$
 - * *alkalmazó*: $\frac{\Gamma \vdash A \quad \Gamma \vdash A \equiv B}{\Gamma \vdash B}$

1.2. Operációs rendszerek

1.2.1. Operációs rendszerek fogalma, felépítése, osztályozásuk.

Operációs rendszerek fogalma Egy program, amely közvetítő szerepet játszik a számítógép felhasználója és a számítógéphardver között. Az operációs rendszer feladata, hogy a felhasználónak egy olyan egyenértékű kiterjesztett vagy virtuális gépet nyújtson, amelyiket egyszerűbb programozni, mint a mögöttes hardvert

Operációs rendszerek felépítése Az operációs rendszerek alapvetően három részre bonthatók:

- a felhasználói felület (a shell, amely lehet egy grafikus felület, vagy egy szöveges)
- alacsony szintű segédprogramok
- kernel (mag), amely közvetlenül a hardverrel áll kapcsolatban.

Operációs rendszerek osztályozása

1. Az operációs rendszer alatti hardver "mérete" szerint:
 - mikroszámítógépek operációs rendszerei
 - kisszámítógépek, esetleg munkaállomások operációs rendszerei
 - nagygépek (Main Frame Computers, Super Computers) operációs rendszerei
2. A kapcsolattartás típusa szerint:
 - kötegelt feldolgozású operációs rendszerek vezérlőkártyás kapcsolattartással
 - interaktív operációs rendszerek.
3. cél szerint: általános felhasználású vagy céloperációs rendszer
4. a processzkezelés: single-tasking, multi-tasking
5. a felhasználók száma szerint: single, multi
6. CPU-idő kiosztása szerint: szekvenciális, megszakítás vezérelt, event-polling, time-sharing
7. a memóriakezelés megoldása szerint: valós és virtuális címzésű

1.2.2. Az operációs rendszerek jellemzése (komponensei és funkciói).

Operációs rendszerek komponensei:

Eszközkezelők (Device Driver) Felhasználók elől el fedik a perifériák különbségeit, egységes kezelői felületet kell biztosítani.

Megszakítás kezelés (Interrupt Handling) Alkalmas perifériák felől érkező kiszolgálási igények fogadására, megfelelő ellátására.

Rendszerhívás, válasz (System Call, Reply) az operációs rendszer magjának ki kell szolgálnia a felhasználói alkalmazások (programok) erőforrások iránti igényeit úgy, hogy azok lehetőleg észre se vegyék azt, hogy nem közvetlenül használják a perifériákat← programok által kiadott rendszerhívások, melyekre rendszermag válaszokat küldhet.

Erőforrás kezelés (Resource Management) Az egyes eszközök közös használatából származó konfliktusokat meg kell előznie, vagy bekövetkezésük esetén fel kell oldania.

Processzor ütemezés (CPU Scheduling) Az operációs rendszerek ütemező funkciójának a várakozó munkák között valamilyen stratégia alapján el kell osztani a processzor idejét, illetve vezérelnie kell a munkák közötti átkapcsolási folyamatot.

Memóriakezelés (Memory Management) Gazdálkodnia kell a memóriával, fel kell osztania azt a munkák között úgy, hogy azok egymást se zavarhassák, és az operációs rendszerben se tegyenek kárt.

Állomány- és lemezkezelés (File and Disk Management) Rendet kell tartania a hosszabb távra megőrzendő állományok között.

Felhasználói felület (User Interface) A parancsnyelveket feldolgozó monito utódja, fejlettebb változata, melynek segítségével a felhasználó közölni tudja a rendszermaggal kívánságait, illetve annak állapotáról információt szerezhet.

Operációs rendszerek funkciói:

Folyamatkezelés A folyamat egy végrehajtás alatt álló program. Hogy feladatát ellássa erőforrásokra van szüksége (processzor idő, memória, állományok I/O berendezések). Az operációs rendszer feladata:

- Folyamatok létrehozása és törlése
- Folyamatok felfüggesztése és újraindítása
- Eszközök biztosítása a folyamatok kommunikációjához és szinkronizációjához.

Memória (főtár) kezelés Bájtokból álló tömbnek tekinthető, amelyet a CPU és az I/O közösen használ. Tartalma törlődik rendszerkikapcsoláskor és rendszerhibáknál. Az operációs rendszer feladata:

- Nyilvántartani, hogy az operatív memória melyik részét ki (mi) használja.
- Eldönteni melyik folyamatot kell betölteni, ha memória felszabadul.
- Szükség szerint allokálni és felszabadítani a memória területeket a szükségleteknek megfelelően.

Másodlagos tárkezelés Nem törlődik, és elég nagy hogy minden programot tároljon. A merevlemez a legelterjedtebb formája. Az operációs rendszer feladata:

- Szabadhely kezelés.
- Tárhozzárendelés.
- Lemez elosztás.

I/O rendszerkezelés • Puffer rendszer.

- Általános készülék meghajtó (device driver) interface.
- Speciális készülék meghajtó programok.

Fájlkezelés Egy fájl kapcsolódó információk együttese, amelyet a létrehozója definiál. Általában program és adatfájlokról beszélünk. Az operációs rendszer feladata:

- Fájlok és könyvtárak létrehozás és törlése.
- Fájlokkal és könyvtárakkal történő alapmanipuláció.
- Fájlok leképezése a másodlagos tárra, valamilyen nem törlődő, stabil adathordozóra.

Védelmi rendszer Olyan mechanizmus, mely az erőforrásokhoz való hozzá férést felügyeli. Az operációs rendszer feladata:

- Különbséget tenni jogos (authorizált) és jogtalan használat között.
- Specifikálni az alkalmazandó kontrolt.
- Korlátozó eszközöket szolgáltatni.

Hálózat elérés támogatása Az elosztott rendszer processzorok adat és vezérlő vonallal összekapcsolt együttese, ahol a memória és az óra nem közös. Adat- és vezérlővonal segítségével történik a kommunikáció. Az elosztott rendszer a felhasználóknak különböző osztott erőforrások elérését teszi lehetővé, mely lehetővé teszi:

- a számítások felgyorsítását,
- a jobb adatelérhetőséget,
- a nagyobb megbízhatóságot.

Parancs interpreter alrendszer Az operációs rendszernek sok parancsot vezérlő utasítás formájában lehet megadni. Vezérlő utasítások minden területhez tartoznak (folyamatok, I/O kezelés...). Az operációs rendszernek azt a programját, amelyik a vezérlő utasítást beolvassa és interpretálja a rendszertől függően más és más módon nevezhetik:

- Vezérlő kártya interpreter.
- Parancs sor interpreter (command line).
- Héj (burok, shell)

1.2.3. A rendszeradminisztráció, fejlesztői és alkalmazói támogatás eszközei.

Rendszeradminisztráció Magának az operációs rendszernek a működtetésével kapcsolatos funkciók. Ezek közvetlenül semmire sem használhatók, csak a hardverlehetőségek kibővítését célozzák, illetve a hardver kezelését teszik kényelmesebbé. A rendszeradminisztráción belül a következő *összetett funkciókat* jelölhetjük ki:

1. **processzorütemezés:** a CPU-idő szétosztása a rendszer- és a felhasználói feladatok (taszkok, folyamatok) között;
2. **megszakításkezelés:** a hardver-szoftver megszakításkérések elemzése, állapotmentés, a kezelőprogram hívása;
3. **szinkronizálás:** az események és az erőforrásigények várakozási sorokba állítása;
4. **folyamatvezérlés:** a programok indítása és a programok közötti kapcsolatok szervezése;
5. **tárkezelés:** a főtár, – mint kiemelten kezelt erőforrás, – elosztása;
6. **periféria-kezelés:** a bemeneti/kimeneti (B/K ill. I/O) igények sorba állítása és kielégítése;
7. **adatkezelés:** az adatállományokon végzett műveletek segítése (létrehozás, nyitás, zárás, írás, olvasás stb.);
8. **működés-nyilvántartás:** a hardver hibastatisztika vezetése és a számlaadatok feljegyzése;
9. **operátori interfész:** a kapcsolattartás az üzemeltetővel.

A konkrét operációs rendszerek a funkciókat másképpen oszthatják fel. Így például az IBM OS operációs rendszerek változataiban négy fő funkciót szoktak megkülönböztetni:

1. a munkakezelést,
2. a taszkkezelést,
3. az adatkezelést és
4. a rendszerstatisztikát.

A rendszeradminisztrációs funkciókat a **rendszermag** valósítja meg, amelynek a szolgáltatásait a már említett rendszerhívásokkal érhetjük el.

Programfejlesztési támogatás fő funkciói:

1. **rendszerhívások:** a programokból alacsony szintű operációsrendszeri funkciók aktivizálására,
2. **szövegszerkesztők:** a programok és dokumentációk írására,
3. **programnyelvi eszközök:** fordítóprogramok és interpreterek (értelmezők) a nyelvek fordítására vagy értelmezésére,
4. **szerkesztő- és betöltő-programok:** a programmodulok összefűzésére illetve tárba töltésére (végcímzés),
5. **programkönyvtári funkciók:** a különböző programkönyvtárak használatára,
6. **nyomkövetési rendszer:** a programok belövésére.

Alkalmazói támogatás Az alkalmazói támogatás funkciói a számítógépes rendszer több szintjén valósulnak meg, és az alábbi fő funkciókra bonthatók:

1. **operátori parancsnyelvi rendszer:** a számítógép géptermi üzemvitelének támogatására;
2. **munkavezérlő parancsnyelvi rendszer:** a számítógép alkalmazói szintű igénybevételének megfogalmazására;
3. **rendszer szolgáltatások:** az operációs rendszer magjával közvetlenül meg nem oldható rendszerfeladatokra;
4. **segéd-programkészlet:** rutinfeladatok megoldására;
5. **alkalmazói programkészlet:** az alkalmazásfüggő feladatok megoldására

1.3. Magas szintű Programozási nyelvek

1.3.1. Adattípus, konstans, változó, kifejezés.

1.3.2. Paraméterkiértékelés, paraméterátadás.

1.3.3. Hatáskör, névterek, élettartam.

1.3.4. Fordítási egységek, kivételkezelés.

1.4. Magas Sintű programozási nyelvek 2

1.4.1. Speciális programnyelvi eszközök.

1.4.2. Az objektumorientált programozás eszközei és jelentősége.

1.4.3. Funkcionális és logikai programozás.

1.5. Adatszerkezetek és algoritmusok

1.5.1. Adatszerkezetek reprezentációja.

1.5.2. Műveletek adatszerkezetekkel.

1.5.3. Adatszerkezetek osztályozása és jellemzésük.

1.5.4. Szekvenciális adatszerkezetek: sor, verem, lista, sztring.

1.5.5. Egyszerű és összetett állományszerkezetek.

1.6. Adatbázisrendszerek

1.6.1. Relációs, ER és objektumorientált modellek jellemzése.

1.6.2. Adatbázisrendszer.

1.6.3. Funkcionális függés.

1.6.4. Relációalgebra és relációkalkulus.

1.6.5. Az SQL.

1.7. Hálózati architektúrák

1.7.1. Az ISO OSI hivatkozási modell.

1.7.2. Ethernet szabványok.

1.7.3. A hálózati réteg forgalomirányító mechanizmusai.

1.7.4. Az internet hálózati protokollok, legfontosabb szabványok és szolgáltatások.

1.8. Fizika 1

1.8.1. Fizikai fogalmak, mennyiségek.

1.8.2. Impulzus, impulzusmomentum.

1.8.3. Newton törvényei.

1.8.4. Munkatétel.

1.8.5. Az I. és II. főtételek.

1.8.6. A kinetikus gázmodell.

1.9. Fizika 2

1.9.1. Elektromos alapfogalmak és alapjelenségek.

1.9.2. Ohm-törvény.

1.9.3. A mágneses tér tulajdonságai.

1.9.4. Elektromágneses hullámok.

1.9.5. A Bohr-féle atommodell.

1.9.6. A radioaktív sugárzás alapvető tulajdonságai.

1.10. Elektronika 1, 2

1.10.1. Passzív áramköri elemek tulajdonságai, RC és RLC hálózatok.

1.10.2. Diszkrét félvezető eszközök, aktív áramköri elemek, alapkapcsolások.

1.10.3. Integrált műveleti erősítők.

1.10.4. Tápegységek.

1.10.5. Mérőműszerek.

1.11. Digitális Technika

1.11.1. Logikai függvények kapcsolástechnikai megvalósítása.

1.11.2. Digitális áramköri családok jellemzői(TTL, CMOS, NMOS).

1.11.3. Különböző áramköri családok csatlakoztatása.

1.11.4. Kombinációs és szekvenciális hálózatok. A/D és D/A átalakítók.

2. Infokommunikációs hálózatok specializáció

2.1. Távközlő hálózatok

2.1.1. Fizikai jelátviteli közegek.

2.1.2. Forráskódolás, csatornakódolás és moduláció.

2.1.3. Csatornafelosztás és multiplexelési technikák.

2.1.4. Vezetékes és a mobil távközlő hálózatok.

2.1.5. Műholdas kommunikáció és helymeghatározás.

2.2. Hálózatok hatékonyságanalízise

2.2.1. Markov-láncok, születési-kihalási folyamatok.

2.2.2. A legalapvetőbb sorbanállási rendszerek vizsgálata.

2.2.3. A rendszerjellemzők meghatározásának módszerei, meghatározásuk számítógépes támogatása.

2.3. Adatbiztonság

2.3.1. Fizikai, ügyviteli és algoritmusos adatvédelem, az informatikai biztonság szabályozása.

2.3.2. Kriptográfiai alapfogalmak.

2.3.3. Klasszikus titkosító módszerek.

2.3.4. Digitális aláírás, a DSA protokoll.

2.4. A RIP protokoll működése és paramétereinek beállítása (konfigurációja).

2.5. Bevezetés a Cisco eszközök programozásába 1

2.5.1. A forgalomszűrés, forgalomszabályozás (Trafficfiltering, ACL) céljai és beállítása (konfigurációja) egy választott példa alapján.

2.6. Bevezetés a Cisco eszközök programozásába 2

2.6.1. A forgalomirányítási táblázatok felépítése, statikus és dinamikus routing összehasonlítása.

Tárgymutató

atomi formula, 5

prímformula, 5