

# PROGRAMOZÁSMÓDSZERTAN 4. FÉLÉV

## TEMATIKA

### ÁLTALÁNOS KÉRDÉSEK

Követelmények az *aláírásért*:

- 2 zárthelyi (✍) megírása,
- beadandó feladat érdemi megoldása és (📁) beadása (megvédése),
- a (📁) gépes zárthelyi megírása.

Követelmények a *legalább kettes gyakorlati jegyért*:

- rendszeres gyakorlatra járás (a vizsgaszabályzatban rögzítettek szerint),
- mindkét zárthelyi legalább 2,
- a beadandó feladat legalább 2,
- a gépes zárthelyi legalább 2.

A gyakorlati jegybe a zárthelyik és a házi feladat(ok) eredményei mellett a gyakorlaton való részvétel minősége is beleszámít.

Ha a(z „elméleti” és géptermi) zárthelyik valamelyikét nem írta meg legalább elégségesre, akkor a pótzárthelyi(ke)t meg **kell** írnia!

A beadandóval szemben támasztott feltétel, hogy

1. a program legyen képes rövid (2-3 *oldalnyi!*) nyitó **tájékoztató** megjelenítésére,
2. mivel a megoldáshoz néhány „komplex” adatszerkezetre van szükség, ezért ezt (ezeket) önálló modulban (Unit-ban és/vagy, Include-állományban), elkülönítve kell megvalósítani. Azaz a program algoritmus is és a kódja is **legalább két-két független program-egységet** alkot.

A beadandó feladat beadásáról (🚫):

1. A meghatározott **formai feltételeknek** eleget nem tevő dolgozatokat nem értékelünk (azt újra be kell adni helyesen; azonban a határidő nem módosul),
2. A beadás **e-mail**-ben történik (ennek dátuma a meghatározó); a dokumentációt ettől függetlenül papíron is beadható (ami azonban nem helyettesíti az e-mail-es beadást!), ekkor kaphat *részletes értékelést* munkájáról.
3. Késedelmes beadás esetén a jegyet hetente 1 jeggyel csökkentjük, de legfeljebb 3 jegy levonásnyi késés lehetséges. 3 hét késés után nem fogadjuk el a beadandót, ez esetben az előadó aláírást nem adhat.
4. A beadandóval kapcsolatos információk **további részleteihez** (pl. értékelési szempontok) olvassa el a beadandóval kapcsolatos információkat: [RekBeaErt.doc](#)-ban, [html](#)-ben; [pdf](#)-ben; és a mintadokumentációt a [doc](#)-ban vagy [pdf](#)-ben.

A beadandókkal szemben támasztott további feltétel, hogy a felhasználó (ez esetben a tesztelő) **választása** szerint

1. az input akár klaviatúra, akár sima TEXT (-fájl) lehet,

2. az output akár az alkalmazás valamely ablaka, akár sima TEXT (-fájl) lehet.

A beadandók névhez rendelése:






1	Árva Zoltán	16	Kmecsek Viktória	11	Reiner Zita
2	Bugyi Sándor	17	Kocsis Bernadett	12	Sárány Róbert
3	Csányi Tibor	18	Kovács Péter Norbert	13	Schoffhauzer Péter
4	Csikfalvi Attila	19	Marosi Polla Ágnes	14	Seprődi Máté
5	Fincza András József	20	Marosi Zoltán	15	Sereghy Zsófia
6	Furkó Árpád Gergely	1	Márton Balázs	16	Stadler János Kolos
7	Gedeon Veronika	2	Micheller Mária	17	Szatmári Boglárka Etelka
8	Hegedüs László	3	Molnár József	18	Szőke Szabolcs Tamás
9	Hubai Szabolcs László	4	Morandini Kristóf Márk	19	Takács Bence
10	Hutóczki Erika	5	Müller Katalin	20	Takács Valéria
11	Illés Márk	6	Nagy Balázs Tamás	1	Váczai Viktória
12	Juhos Tamás	7	Németh Gabriella	2	Vári Erika
13	Kálcza Tamás	8	Németh Katalin	3	Várkonyi Teréz Anna
14	Kiss Attila	9	Ötvös Petra		
15	Kiss Endre Farkas	10	Ráczpali István		

## RÉSZLETES TEMATIKA

A tematikában elsősorban az alábbi irodalmakra építünk és hivatkozunk:

1. Módszeres programozás – Rekurzió (*μlógia* 4, 4. kiadás),
2. Módszeres programozás – Rekurzív típusok (*μlógia* 27, 1. kiadás),
3. Varga: Programok analízise és szintézise (Akadémiai Kiadó),
4. Módszeres programozás – Gráfok, gráfalgoritmusok (*μlógia* 38, 2. kiadás),
5. Párhuzamos programok szintézise (<http://izzo.inf.elte.hu/szlavi/ProgModsz/SzinkEgyesit.zip>)
6. Cormen et al.: Algoritmusok (Műszaki Könyvkiadó, 1. kiadás)
7. Módszeres programozás – Hatékonyság (*μlógia* 6, 4. kiadás)
8. Kozma-Varga: Adattípusok osztálya (Informatika Tanszékcsoport)
9. Rónyai-Ivanyos-Szabó: Algoritmusok (TYPOT<sub>EX</sub>)
10. Szlávi: Gondolatok a típus-specifikációk kompatibilitásának vizsgálatáról (*μlógia* Szilánkok 11)

Sorszám	Előadás	Gyakorlat
<a href="#">1</a> 02.16.	<b>Rekurzió 1.:</b> A rekurzió fogalma, példák, a megvalósítás problémái. [1/6-25, 9/42-44]	<b>Mohó algoritmusok. 1.</b> <b>Mohó algoritmusok. 2.</b>
<a href="#">2</a> 02.23.	<b>Rekurzió 2.:</b> A rekurzió és az iteráció kapcsolata. [1/31-46, 2/68-74]	<b>Rekurzív függvények és eljárások</b> (és iteratív párjaik), hatékonyságvizsgálat. <b>QuickSort</b> rendezés rekurzív algoritmusának elemzése.
<a href="#">3</a> 03.02.	<b>Rekurzió 3.:</b> A rekurzió mint típuskonstrukciós eszköz. [2/5-26]	<b>QuickSort</b> fél-iteratív és iteratív változatai. <b>Programozási tételek</b> rekurzív és iteratív algoritmusai (+specifikálás).
<a href="#">4</a> 03.09.	<b>Rekurzió 4.:</b> Bináris fák – kereső és rendező fák. [2/27-43, 46-54, 6/208-225, 115-126, 9/57-56]	<b>Programozási tételek</b> rekurzív és iteratív megvalósítása (kódolás). <b>BinFa modul</b> megvalósítása.

5 03.16.	<b>Rekurzió 5.:</b> Általános fák,ábrázolásai, kiegyensúlyozott fák, B-fák. [2/44-45,55-62, 6/327-344, 9/69-79]	<b>Kereső fák</b> kezelése a BinFa modulra építve. Bináris fák <b>kiegyensúlyozása</b> . 1.
6 03.23. 	<b>Dinamikus programozás</b> [7/43-45, 6/259-282, 9/302-305]	Bináris fák <b>kiegyensúlyozása</b> . 2. <b>HeapSort</b> . Hatékonyságvizsgálat.
7 03.30.	<b>Gráfok 1.:</b> Gráfábrázolások, <b>Gráfok 2.:</b> a gráf mint típuskonstrukciós eszköz. [4/5-33, 6/404-406]	<b>Dinamikus programozás</b> . 1. <b>Dinamikus programozás</b> . 2.
8 04.06.	<b>1. zárthelyi</b> ✍  <a href="#">önálló feladat kiosztása (pdf)</a>	<b>Gráfábrázolások</b> . „Statikus” gráf modul megvalósítása. <b>Gráfábrázolások</b> . „Statikus” gráf modul megvalósítása. (Konverziók a különféle ábrázolások között.)
	<i>Tavaszi szünet</i>	<i>Tavaszi szünet</i>
9 04.20.	<b>Gráfok 3.:</b> Gráf-bejárési algoritmusok. Útke-resések. [4/27-38, 6/407-420 ,444-463, 9/113-119]	 <b>gépes zárthelyi (Lovarda)</b> <b>04.19. 11-14 (1,2. csop., a feladat),</b> <b>04.21. 8-11 (3-5. csop., a feladat)</b>
10 04.27.	<b>Gráfok 4.:</b> Gráf összefüggőség-vizsgálata. Fe-sztítőfák. Hálózati folyamatok. [4/43-53,58-66, 6/477-489, 9/122-124, 151-161]	<b>Szélességi bejárás.</b> Szélességi bejárás alkalmazásai 1. <b>Szélességi bejárás</b> alkalmazásai 2.
11 05.04.	<b>Programhelyesség.</b> Strukturált programozás, programhelyesség, típushelyesség. [3/134-167, 8/67-79, 10]	<b>Mélységi bejárás.</b> Mélységi bejárás alkalmazásai 1. <b>Mélységi bejárás</b> alkalmazásai 2.
12 05.11.	<b>Eötvös-Nap – az előadás orvul elmarad!!!</b>  <b>önálló feladat beadása</b> 📄	<b>Gráf-összefüggőség.</b> Eötvös-/Pázmány-Nap – elmaradó gyakorlatok
13 05.18.	<b>2. zárthelyi</b> ✍	<b>Komplex gráfos feladat.</b> <b>Programhelyesség.</b> Egy konkrét feladat bizonyítása; Böhm-Jacopini tétel bizonyítása.
14 05.25.	<b>Pót papiros zárthelyik (05.25.)</b> ✍	 <b>Pót gépes zárthelyi 05.26. 8-11 (Lovarda)</b>

## VIZSGÁK

Gyakorlati jegy UV: 05.30. (10 óra, Nyelvi Labor)

Szigorlat: 06.02. (10 óra), 06.08. (8 óra), 06.14. (8 óra), 06.22. (8 óra), 06.28. (8 óra)

[Szigorlati tematika](#) ()