

PROGRAMOZÁSMÓDSZERTAN 3. FÉLÉV

TEMATIKA

ÁLTALÁNOS KÉRDÉSEK

Követelmények az aláírásért:

- 2 zárthelyi (✍) megírása,
- beadandó feladat (📄) beadása (és esetleges 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 programot 4GL környezetben kell fejleszteni,
2. a program legyen képes –a felhasználó kérésére, pl. menü keresztül kiválasztható– lényegre törő, információdús **tájékoztató** megjelenítésére,
3. továbbá akár fájlból, akár klaviatúráról lehessen paraméterezni, az eredményt –szintén a felhasználó választása szerint– akár fájlba, akár az alkalmazás ablakában megjeleníteni,
4. a program helyes működésének dokumentálásához szükséges jónéhány (legalább 3) **teszt-adatsor**, fájlokban. Ezeket a végrehajtható kód mellett kell elhelyezni az anyagban.

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); az e-mail tárgya legyen: „**Beadandó**”; 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 4 vagy több heti késés esetén a beadandót nem fogadjuk el, így a tárgy aláírását megtagadjuk;
4. A feladatok névhez rendelése:

Név	Feladatsorszám
Ajtai Veronika	1
Árva Zoltán	11
Balaton Marcell Balázs	1

Bársony Attila	2
Bernát Péter	1
Bor József	12
Borvendég Orsolya	13
Bugyi Sándor	2
Csányi Tibor	3
Csikfalvi Attila	1
Dankházi Gergely Mátyás	2
Duplinszki Miklós	2
Fajth István	14
Fehér Dániel	3
Fincza András József	3
Furkó Árpád Gergely	3
Gedeon Veronika	4
Harmath Zsolt	5
Hegedüs László	6
Horváth Mária	7
Hubai Szabolcs László	4
Hutóczki Erika	14
Juhos Tamás	15
Kálcza Tamás	16
Kiss Attila	8
Kiss Endre Farkas	17
Kmeecs Viktória	9
Kovács Péter Norbert	15
Marosi Polla Ágnes	18
Marosi Zoltán	4
Márton Balázs	4
Micheller Mária	5
Molnár József	10
Morandini Kristóf Márk	6
Müller Katalin	5
Nagy Balázs Tamás	1
Németh Gabriella	6
Németh Katalin	5
Ötvös Petra	16
Palkó Tamás Attila	7
Pápa Éva Orsolya	2
Ráczpali István	7
Reiner Zita	8
Schoffhauzer Péter	9
Seprődi Máté	11
Sereghy Zsófia	10
Stadler János Kolos	11
Szabó Krisztina	8
Szatmári Boglárka Etelka	12
Szegedi Atina	9
Szőke Szabolcs Tamás	6
Szűcs Tibor	7
Takács Bence	3
Takács Valéria	10
Váczi Viktória	17




Vári Erika	12
Várkonyi Teréz Anna	13
Varsányi Anita Krisztina	13
Volszky Péter	8
Zolnai László	4







RÉSZLETES TEMATIKA

A tematikában elsősorban az alábbi irodalmakra építünk és hivatkozunk:

1. Módszeres programozás – Hatékonyság (*μlógia* 6, 3. kadás),
2. Módszeres programozás – Adatfeldolgozás (*μlógia* 12, 4. kadás)
3. Módszeres programozás – Szövegfeldolgozás (*μlógia* 14, 3. kadás)
4. Cormen et al.: Algoritmusok (Műszaki Könyvkiadó)
5. Adatfeldolgozás. Tömörített anyag az előadásról. ([ZIP](#))
6. Programok – programszempifikációk (<http://izzo.inf.elte.hu/szlavi/ProgModsz/Progspec.pdf>)
7. Első lépések a Lazarus fejlesztői környezetben ([PDF](#))

A 3 évvel ezelőtti [megjegyzéseimet](#), amelyet az akkori 1. zárthelyi után írtam, érdemes elolvasni és megfontolni! ([pdf](#))

Sorszám dátum	Előadás	Gyakorlat
1 09.15. 	Programozási folyamat: áttérés egy 4GL programfejlesztői környezetre	Adminisztráció 4GL 1.. egy egyszerű inputú (edit, button...) program fejlesztése FreePascal-Lazarus (FPL) környezetben. (Lásd 7.)
2 09.22.	Szövegfeldolgozás 1.: Szövegtípusok értelmezései, modulok [3/5-28]	4GL 2.. összetett inputú (memo, stringGrid) program fejlesztése FPL környezetben. 4GL 3.. menüvezérelt program fejlesztése FPL környezetben.
3 09.29. 	Szövegfeldolgozás 2.: Szövegtípusok speciális feladatai: formázás, szűrés, tömörítés [3/29-46] önálló feladatok	4GL 2.. tájékoztatóval rendelkező program (újabb ablak, szövegfájl) fejlesztése FPL környezetben. Karakter-, szövegmodul készítése (ékezetes betűk).
4 10.06.	Szövegfeldolgozás 3.: Szövegtípusok speciális feladatai: keresések, titkosítás, hipertext [3/29-46] + [4/185-206]	Szövegmodul továbbépítése (sor, lap, formázások). Szövegmodul továbbépítése (keresés).
5 10.13.	Adatfeldolgozás 1.: Adatfeldolgozási feladatok, struktúrafinomítás [2/41-55] + <i>program példa a finomításhoz</i>	Szövegtranszformátor 1.: sorigazítások. Szövegtranszformátor 2.: szűrések.
6 10.20.	Adatfeldolgozás 2.: Struktúramegfeleltetés konfliktusai [2/56-67]	Makroprocesszor 1.: tervezés. Makroprocesszor 2.: megvalósítás.
7 10.27.	Adatfeldolgozás 3.: Időszerűsítés, külső rendezések [2/68-80]	Makroprocesszor 3.: megvalósítás (folytatás). Struktúramegfeleltetéses példák 1.: (adatbevitel űrlappal; majd) lapozott listázás.
8	ősi szünet	
9 11.10.	1. zárthelyi 	Struktúramegfeleltetéses példák 2.: összegfokozatos listázás . Adatfeldolgozási feladatsor.

10 11.17.	Hatékonyság 1.: Végrehajtási idő csökkentése 1. [1/7-45]+ <i>alapfogalmak (Ordó ...)</i> [4/18-26]	Adatfeldolgozási feladatsor (folytatás).  gépes zárthelyi (11.17. 12⁰⁰-15⁰⁰)
11 11.24.	Hatékonyság 2.: Végrehajtási idő csökkentése 2.[1/46-74]  önálló feladat beadása 	Gyakoriság szerinti elrendezés hatékonyságnövelő szerepének vizsgálata. (<u>L.</u>) Hatékonyság 1.: 1. feladatsor (globális/idő).
12 12.01.	Hatékonyság 3.: Helyfoglalás csökkentése [1/75-107] Hatékonyság 4.: Bonyolultság csökkentése [1/108-129] 	Hatékonyság 2.: 2. feladatsor (globális/idő). Hatékonyság 3.: 3. feladatsor (globális/hely=elemméret-, sorozathossz-csökkentés).
13 12.08. 	Mohó algoritmusok [4/283-305]	Hatékonyság 4.: a 3. feladatsor elemméret-csökkentése részfeladatsor kódolása. Hatékonyság 5.: 4. feladatsor (lokális=kód) .
14 12.15.	2. zárthely ✍	Mohó algoritmusok. Mohó algoritmusok (folytatás).
+1 tartalék	Programozási folyamat: Programspecifikációs eszközök (VDM, Z ...) [6]	Tartalék Tartalék
	Pótpapíros zárthelyik (2005.12.22. 8³⁰-11) ✍	 Pót, gépes zárthelyi (2005.12.22. 11⁰⁰-14⁰⁰)