

Bevezető

- A programozás alapjai 2 (VIII AA03)
- Két fontos 3. féléves tárgyhöz is alapozunk:
 - Szoftvertechnológia
 - Programozás alapjai 3.
- OOP alapjai a C++ segítségével
 - nem a C++ nyelv a lényeg, hanem az objektum orientáltság
- Előző félév C ismeretére építünk
- minimális UNIX, + fejl. eszközök, hogy ismerjék, tudják használni (hftest)

Bevezető /2

- A félév során a tárgyra fordítandó idő 180 óra (45')! Ebből a kontakt órák száma 56.
- Önálló munka: 124 óra, azaz átlagosan heti $8.8 \times 45' = 6.6$ óra önálló gyakorlás, készülés szükséges a tárgy elvégzéséhez.
- Nem meglepő, ha a heti ellenőrző feladatok némelyike 2 óra önálló munkát is igényel.
- Másolással, mások megoldásának beadásával nem az oktatókat csapják be, hanem saját magukat.

Gyakorlás, gyakorlás, ...

A tantárgy elvégzéséhez átlagosan szükséges tanulmányi munka:

Kontakt óra	56
Felkészülés előadásokra	14
Felkészülés laborra	28
Felkészülés zárthelyikre	43
Felkészülés beszámoló dolgozatokra	9
Házi feladat elkészítése	30
Kijelölt írásos tananyag elsajátítása	0
Vizsgafelkészülés	0
Összesen	180

Bevezető /3

- Minden laboron ellenőrizzük a felkészültséget:
 - Labor előtt beadandó ellenőrző feladatokkal. **Határidő** a laborfoglalkozást megelőző **kedd reggel 6 óra**.
 - Labor első perceiben ellenőrző dolgozatot íratunk,
- A laborok nagy részében skeleton programokat kell kiegészíteni. Ezek a tanult anyagrészre fókuszálnak.
- A heti előadásokhoz javasolt kiegészítő anyagok komplexebb példákat tartalmaznak.
- Előadásra érdemes a diákat előre kinyomtatni, mert sokszor igen nagy az anyag és gyorsan haladunk.

Bevezető /4

- Tárgyhoz kapcsolódó hivatalos információk:
 - infocpp.iit.bme.hu
 - FONTOS: hivatalos e-mail beállítása a Neptunban
 - A laborok beosztásán a második hét végéig lehet változtatni a létszámkereteken belül (→ csere) az érintett laborvezetők hozzájárulásával.
- Irodalom:
 - Adatlapon megadottakból bármi. Javaslat:
Szoftverfejlesztés C++ nyelven, SZAK, 2007

Követelmények

- A TAD: <https://portal.vik.bme.hu/kepzes/targyak/VIIIAA03>
- VIIIAA03: 2+0+2 félév végi jegy, 6 kredit
 - A 6 kredit megszerzése összesen 180 óra (45') befektetést igényel
 - jelenlét (külön-külön 70%)
 - 12 db e.f. (ellenőrző feladat) a labor előtt (1-1p)
 - 12 db e.d. (ellenőrző dolgozat \approx beugró) laboron (1-1p)
 - A fentiekből legalább 16 pontot (2/3-ot) kell elérni.
 - 3 db i.b. (írásos beszámoló \approx kisZH) laboron (10-10p)
 - A két legjobbnak külön-külön is el kell érnie a 40%-ot (4p)
 - 2 db ö.é. (összegző értékelés \approx nZH) külön időpontban
 - kettőnek együtt el kell érni 50%-ot. (40 pontot)
 - 1 db h.f. (házi feladat)
 - elfogadása a jegy feltétele, extra pont szereshető a határidők betartásával (6p)
 - jegy: a két legjobb i.b. (20p) + a két ö.é. (80p) + extra (6+7p)

Hiányzás, pótlás

- Hiányzás max. 30% (max 4 hiányzás)
- e.d. és i.b. (labor és kisZH) nem pótolható
- HF vizsgaidőszakban nem pótolható
 - pótlási héten is csak akkor, ha volt előzménye
- Házi minél előbb érdemes elkezdeni.
- Házi egy része generikus, ami a 7. hét körül lesz, előadáson de érdemes úgy hozzáfogni, mintha nem lenne generikus.

Nulladik ZH-val kapható kedvezmény

- A jelentkezés Jelentkezés - 0. ZH (jporta) feladat határidőre történő megoldásával.
- A ZH időpontja: 02.18. 18:15-20:00
- Kedvezmény (nem kötelező elfogadni):
 - Labor és előadás jelenlét a teljes félévre.
 - kZH pontok: dolgozat eredménye alapján.
pl. 76% → 7.6 → 8 pontot kap.
- NZH és HF követelményét teljesíteni kell.

Extra pontok

- Az extra pontok begyűjtése nem kötelező, de megszerzésük 1 jeggyel jobb osztályzatot eredményez.
- HF részfeladatainak határidőre történő leadásával, $1+1+2+2=6$ extra pontot lehet szerezni, feltéve, ha a véghatáridő teljesül.
- További 7 extra pontot lehet szerezni a hftest feladatok megoldásával <https://infocpp.iit.bme.hu/hftest>
- Korábbi években az első pont a feladat lekérdezéséért járt, ez most **megszűnt**. Csak helyes megoldásért jár pont.

hfctest példa/1

```
ural2:~$ mkdir bin
```

```
PATH=$PATH:~/bin
```

```
ural2:~$ ln -s ~szebi/hfctest bin
```

```
ural2:~$ mkdir work
```

```
ural2:~$ cd work
```

```
ural2:~/work$ pico f1.c
```

```
# példa program copy-paste
```

```
# innen: https://infocpp.iit.bme.hu/hfctest\_feladat\_tipusok
```

```
echo 'PATH=$PATH:~/bin' >> ~/.bash_login
```

Célszerű később
betenni a
.bash_login-ba

hftest példa/2

```
#include <stdio.h>  
char myid[ ] =  
    "Szeberenyi Imre <szebi@iit.bme.hu> b12345\n";  
char mytask[ ] = "Feladat = 1 ISO 8859-2\n";  
int main() {  
    setbuf(stdout, NULL); // kikapcsolja a bufferelést  
    printf(myid);  
    printf(mytask);  
    return(0);  
}
```

hfctest példa/3

ural2:~/work\$ gcc f1.c -o f1

ural2:~/work\$./f1

Szeberenyi Imre <szebi@iit.bme.hu> b12345

Feladat = 1 ISO 8859-2

ural2:~/work\$ ~/bin/hfctest f1 #ha nincs a path-ban

Hello szebi!

Tesztetem a "f1" programot...

Sikeres azonosítás:

Szeberenyi Imre <szebi@iit.bme.hu> B12345

hftest példa/4

2. feladat Szeberenyi Imre (B12345) részére:

Valósítsa meg a következő valós értékű függvényt 1e-4 pontossággal!

$$f(X) = X/80.40, \text{ ha } X > 1,$$

$$f(X) = 0.313*X^4 - 0.627*X^3 + 40.12*X^2 + 1*X - 1, \text{ ha } X \leq 1,$$

A függvényhez írjon programot, amely a standard inputról file végéig valós számokat olvas be, és kiszámítja az ezekhez tartozó függvényértékeket! Az eredményt a standard outputra kiírja, ki úgy, hogy minden beolvasás után azonnal írja ki a függvényértéket!

A teszt eredménye: SIKERES (exit status:0,0) 1: 1/1

Összes hftest pontja: 0

Ha mindent jól csinált, ilyen szöveg jelenik meg.

Enek a feladatnak a megoldását 2. feladatként kell beküldeni