

Szigorlati tematika'2006

Programozásmódszertanból

A tételek elé két alapelv: ♥

1. megméretik a vizsgázó konkrét algoritmusismerete – így minden tételbe bele kell ágyazni néhány jellemző algoritmust és példát;
2. megméretik absztrakciós képessége – így minden tétel tartalmazzon erre utaló jeleket (pl. formalizmust).

Tételek

1. Programozási alapok 1.:

Algoritmizálás: Algoritmikus és alapvető adatszerkezetek; és leírásuk a programban; 'strukturálsruzerinti feldolgozás' elve, strukturált programozás lényege. **Algoritmusleíró eszközök.**

Adattípusok: Adatok jellemzői, a típus fogalma, a típusokhoz kapcsolódó műveletek osztályozása, „szabványos” elemi és összetett típusok; sorozattípusokról általában (ábrázolási lehetőségek, műveletek). Típus algebrai specifikációja. A **modul** mint a típusmegvalósítás kerete. A modul fogalom megjelenése(i) a (Turbo) Pascal-ban, Lazarus-ban.

2. Programozási tételek 1.:

Sorozathoz érték rendelése. Sorozathoz sorozat rendelése. (Tételkimondás: absztrakt feladat + absztrakt algoritmus. Példa.)

3. Programozási tételek 2.:

Sorozatokhoz sorozat rendelése. Sorozathoz sorozatok rendelése. (Tételkimondás: absztrakt feladat + absztrakt algoritmus. Példa.)

4. Programozási tételek 3.:

Programtranszformációk célja, formája, példái; tételek **összeépítése**. (Legalább 2 összetett feladat specifikálása: absztrakt feladat + absztrakt algoritmus + programtranszformációk felhasználása a tételek összeépítése során.)

5. Programozási tételek 4.:

Rendezések. (A rendezésfeladat általános specifikálása; alaprendezések algoritmusai, hatékonysági jellemzés. A módosított keresési feladatoknál specifikáció.)
„**Rendezésmélet**”. (Az elvileg legjobb rendezések jellemzői.)

6. Speciális programozási stratégiák 1.:

Keresések („absztrakt lineáris keresés”, lineáris keresés rendezettben, **backtrack**: algoritmusok + hatékonyságok).

Tételek a visszalépéses keresés elvére építve. (Legalább 2 tétel újrafogalmazása. Tételkimondás: absztrakt feladat + absztrakt algoritmus. Példa.)

7. Típuskonstrukciós eszközök 1.:
Tömbök. (Axiómák, reprezentáció, implementáció; címfüggvények, speciális és hiányosan kitöltött tömbök.)
Listák (axiómák, egy reprezentáció és implementáció.) és alkalmazási **példák** (legalább 2 programozási tétel újrafogalmazása listában ábrázolt sorozatokra).

8. Típuskonstrukciós eszközök 2.:
Vermek. (Axiómák, reprezentációk, implementációk.)
Sorok. (Axiómák, reprezentációk, implementációk.)
Kifejezéskiértékelési módszerek: Rutishauser-módszer, Lengyel-forma.

9. Típuskonstrukciós eszközök 3.:
Prioritási sorok. (Axiómák, reprezentáció, implementáció; alkalmazási példák.)
File-ok. (File-osztályok lényegi és algebrai leírása, exportmoduljaik; megvalósítási problémák, a file-osztályok megvalósításának egy lehetséges terve.)

10. Adatfeldolgozás 1.:
Alapfogalmak: $Z=F(X)$ tétel, a 'struktúraszerinti feldolgozás' elve, típusfinomítás és -megfeleltetés; **konfliktusok** és feloldásaik speciális feltételek teljesülése esetén (példák).

11. Adatfeldolgozás 2.:
Időszerűsítés ($Z=F(X)$ vagy az összefuttatás tétel alapján).
Rendezetlenségi konfliktus általános megoldási módszerei (a 4-, a 3-munkafájlos, a prioritási soros rendezés).

12. Szövegfeldolgozás 1.:
A **szövegtípusok** szűkebb és tágabb értelmezése (karakter, szöveg, karakterfájl típusok exportmodulja, implementációs problémák).
Formázások, szűrések és (egyszerű) tömörítések algoritmusai a típusmegfeleltetés alapján.

13. Szövegfeldolgozás 2.:
Szövegtömörítések (Huffman és LZW-kódolás) algoritmusainak,
szöveg minta-keresési módszerek (Knuth-Morris-Pratt, Boyer-Moor, Rabin-Carp) lényege.

14. Programtechnológia 1. – globális hatékonyság:
A végrehajtási **idő csökkentése** a **ciklus-lépésszám** csökkentése által.
A végrehajtási **idő csökkentése** a **ciklus egyszeri végrehajtási idejének** csökkentése által.

15. Programtechnológia 2. – globális hatékonyság:
Hatékonyság – globális hatékonyság 2.:
A **helyfoglalás** és a **bonyolultság csökkentése**.
Programergonómia: tájékoztató, súgó; menük, ablakok és megvalósítási problémáik.

-
- 16. Programtechnológia 3.:**
Lokális hatékonyság. A hatékonyság mérése.
Tesztelés, hibakeresés, hibakeresési eszközök (általában és egy konkrét nyelvi környezetben).
Kódolás: Pascal nyelv elemei: programszerkezet, adatdeklaráció, algoritmikus szerkezetek, eljárások, függvények; néhány elemi kódtranszformáció. A modul fogalom megvalósítási lehetőségei a (Turbo) Pascal-ban/Lazarus-ban.
-
- 17. Programozási alapok 2.:**
A programozás **folyamata**.
A programozás **alapfogalmai, specifikáció**. Programspecifikációs eszközök [21].
Programozás-elmélet: Strukturált és nem strukturált programok. (Strukturáltsággal kapcsolatos fogalmak, tételek kimondása, Böhm-Jaccopini tétel bizonyítása, bonyolultsági mértékek.)
-
- 18. Rekurzió 1.:**
A rekurzió **megvalósítás** problémái.
A **rekurzió és az iteráció** kapcsolata.
Rekurzív függvények iteratív kiszámítása.
-
- 19. Rekurzió 2.:**
A rekurzió mint **specifikációs eszköz** (2-3 programozási tétellel példázva).
A rekurzió mint **típuskonstrukciós eszköz** (exportmodul, megvalósítási lehetőségek).
-
- 20. Rekurzió 2.:**
Bináris fák – kereső fák (specifikálás...bejárások... kiegyensúlyozás).
Rendező fák (kupac rendezés).
Általános fák (ábrázolási lehetőségek), a **B-fák** lényege, a beillesztés algoritmusai.
-
- 21. Gráfok 1.:**
Gráfábrázolások, a gráf mint típuskonstrukciós eszköz (reprezentációs és implementációs példák).
Gráfbejárási algoritmusok.
-
- 22. Gráfok 2.:**
Gráfbejárások alkalmazásai: Útkeresések (egy út, legrövidebb út, legrövidebb út súlyozott gráfra).
Összefüggőségvizsgálat, feszítőfák.
-
- 23. Speciális programozási stratégiák 2.:**
Mohó algoritmusok. (Elvek, példák)
Dinamikus programozás. (Elvek, példák)
-
- 24. Párhuzamosság:**
Problematika és megoldási ötletek (példákon; korutínok, szemaforok, monitorok, üzenetátadás).
Párhuzamos programok szintézise (védett utasítások megvalósítása szemaforokkal; a 'stafétát átadása' módszer lényege).
-

Irodalom

1. Módszeres programozás – Programozási bevezető (μlógia 18)
2. Módszeres programozás – Programozási tételek (μlógia 19)
3. Wirth: Algoritmusok + adatstruktúrák = programok (Műszaki Könyvkiadó)
4. Módszeres programozás – A programkészítés technológiája (μlógia 21)
5. Varga: Programok analízise és szintézise (Akadémiai Kiadó)
6. Fóthi: Bevezetés a programozáshoz (Tankönyvkiadó)
7. Nievergelt et al: Matematikai problémák megoldásainak számítógépes módszerei (Műszaki Könyvkiadó)
8. Programtranszformációk (<http://digo.inf.elte.hu/~szlavi/ProgModsz/Progtran.htm>,
<http://digo.inf.elte.hu/~szlavi/ProgModsz/Progtran.pdf>)
9. Módszeres programozás – Adattípusok (μlógia 34, 1. kiadás),
10. Előadás a file-típusokról és a táblázattípusról (μlógia Szilánkok 8)
11. Módszeres programozás – Szövegfeldolgozás (μlógia 14)
12. Függvényábrázolás (μlógia 16)
13. Varga: Rendszerprogramok elmélete és gyakorlata (Akadémiai Kiadó)
14. Módszeres programozás – Hatékonyság (μlógia 6)
15. Módszeres programozás – Adatfeldolgozás (μlógia 12)
16. Módszeres programozás – Rekurzió (μlógia 4)
17. Módszeres programozás – Rekurzív típusok (μlógia 27)
18. Módszeres programozás – Gráfok (μlógia 38)
19. Párhuzamos programok szintézise
(<http://digo.inf.elte.hu/~szlavi/ProgModsz/SzinkEgyesit.zip>)
20. Programozási tételek specifikációja – összefoglaló
(<http://digo.inf.elte.hu/~szlavi/ProgModsz/Prtetel.pdf>)
21. Programok – programspecifikációk
(<http://digo.inf.elte.hu/~szlavi/ProgModsz/Progspec.pdf>)
22. Cormen et al.: Algoritmusok (Műszaki Könyvkiadó, 1. kiadás)
23. Rónyai-Ivanyos-Szabó: Algoritmusok (TYPOT_EX)
24. Mohó algoritmusok
(<http://digo.inf.elte.hu/~szlavi/InfoSavaria05/MohoAlgoritmusok.pdf>)
25. Dinamikus programozás
(<http://digo.inf.elte.hu/~szlavi/PrM4felev/PDF/DinamikusProgramozas.pdf>)
26. Az előadások pdf-anyagai ([1. félév](#), [2. félév](#), [3. félév](#), [4. félév](#))

♥ Ezt különösen a „meseszerűnek” tűnő tételek esetében fontos tudatosítani!