Egy-egy feladat a 2000/2001-es Nemes Tihamér Országos Középiskolai Számítástechnikai Verseny döntőjének feladataiból

I. kategória.

Egyesek nagyon szerencsétlen napnak tartják, ha a pénteki nap valamelyik hónap 13. napjára esik. Készíts programot (PENTEK:PAS vagy PENTEK.BAS vagy PENTEK.C), amely egy adott évben megmondja, hogy mely hónapok 13. napja esik péntekre!

A program első lépésként olvassa be, hogy melyik évről van szó, illetve, hogy az év első napja milyen napra esik (HÉTFŐ vagy KEDD vagy ... vagy VASÁRNAP). Ezután írja ki azon hónapok nevét, amelyek 13. napja pénteken van! Figyelj a szökőévekre!

`Példa: Bemenet: 2001  HÉTFŐ    Kimenet: ÁPRILIS JÚLIUS`

II. kategória.

Magyarország folyóiról feljegyeztük, hogy milyen másik folyóba folynak bele (pl. a Rába a Dunába, a Sajó a Tiszába, a Hernád a Sajóba, ...). Minden folyó legfeljebb 1 másikba folyhat bele, de lehet hogy egybe sem (pl. Duna, de a Zala sem folyóba folyik bele).

Csónaktúrákat szeretnénk szervezni, de a könnyebbség kedvéért csak úgy, hogy minden folyón a folyás irányában haladjunk.

Készíts programot (TURA.PAS vagy TURA.C), amely megadja, hogy

A. két különböző folyón indult túra hol találkozhat;

B. az első túrát bevárhat-e egy második úgy, hogy nem indul el addig, amíg az első oda nem ér, és ha igen, akkor az elsőnek hány folyón kell addig haladnia (ha ugyanazon a folyón indulnak, akkor 1, ha az egyik folyó éppen belefolyik a másikba, akkor 2, ...).

A TURA.BE állomány első sorában az az N egész szám van, ahány folyóról tudjuk, hogy melyikbe folyik bele ($1 \leqslant N \leqslant 1000$). A következő $2*N$ sor mindegyike egy-egy folyó nevét tartalmazza, a második, negyedik, ... sor azt hogy melyik folyó, a harmadik, ötödik, ... pedig azt, hogy melyikbe folyik bele. Az utolsó két sorban egy-egy folyó neve van, az első és a második túra kezdete. (A folyónevek legfeljebb 20 betűsek.)

A TURA.KI állomány első sorába annak a folyónak a nevét kell írni, ahol a két túra először találkozhat, a sor legyen üres, ha a két túra Magyarországon nem találkozhat (pl. ha az első a Dunán, a második pedig a Tiszán indult). A második sorba azt az egész számot kell írni, ahány folyón az első túrának át kell haladnia, hogy a második túra kezdetéhez érjen. Ha a második nem tudja bevárni az elsőt, akkor ez a szám 0 legyen!

III. kategória.

A porcelángyár égetőkemencéjéhez futószalagon érkeznek az égetésre váró tárgyak. Minden tárgynak ismert a súlya és a kiégetéséhez minimálisan szükséges idő. Az égetésre váró tárgyakat az érkezésük sorrendjében kell kiégetni. Egyszerre több tárgyat is rakhatunk a kemencébe, az összsúlyuk azonban nem haladhatja meg a kemence kapacitását. Az égetési idő egy menetben mindig a kemencébe rakott tárgyak minimális égetési idejének a maximuma kell legyen.

Készíts olyan programot (KEMENCE.PAS, KEMENCE.C), amely kiszámítja, hogy legkevesebb mennyi idő kell az összes tárgy kiégetéséhez, továbbá megadja azt is, hogy ezen idő eléréséhez mely tárgyakat kell egy-egy menetben a kemencében együtt égetni.

KöMaL – 2001/8 SZT0108 cianmagentayellow2001. november 6. – 7:02 (1. lap)

A KEMENCE.BE állomány első sora két egész számot tartalmaz; a tárgyak N ($1\leqslant N\leqslant 10\,000$) számát és a kemence K ($1\leqslant K\leqslant 1000$) kapacitását. A következő N sor mindegyike két egész számot tartalmaz; egy tárgy S ($1\leqslant S\leqslant 1000$) súlyát és E ($1\leqslant E\leqslant 100$) minimális égetési idejét.

A KEMENCE.KI állomány első sorába az összes tárgy kiégetéséhez minimálisan szükséges időt kell írni. A következő sorok mindegyikébe két egész számot, I-t és J-t kell írni egy szóközzel elválasztva, I az első, J pedig az utolsó tárgy sorszáma, amelyek egyszerre kerülnek a kemencébe.

❄

A Nemes Tihamér OKSzTV-n az 2000/2001-es tanévben az első fordulóban 5033 tanuló vett részt 367 iskolából, közülük 683-an határon túli magyarok. A második, regionális fordulót az ország 30 városában rendeztük, amelyen 759 tanuló vett részt. A harmadik fordulót, az országos döntőt Budapesten rendeztük két helyszínen, amelyen 193 tanuló vett részt.

A II–III. kategória 10–11. osztályosai közül a legjobb 14 számára olimpiai válogatóversenyt rendeztünk, amelynek alapján kiválasztottuk az idei Közép-Európai Informatikai Diákolimpián, Zalaegerszegen résztvevő magyar csapat tagjait.

1. csapat: *Csóka Endre* (Debrecen, Fazekas Mihály Gimnázium); *Hargitai Gábor* (Ócsa, Bolyai János Gimnázium); *Pszota Zsolt* (Vác, Boronkay György Szki); *Pallos Péter* (Fazekas Mihály Fővárosi Gyakorló Gimnázium).

2. csapat: *Borosán Péter* (Zalaegerszeg, Zrínyi Miklós Gimnázium); *Szakál Zoltán* (Keszthely, Vajda János Gimnázium); *Novák Zoltán* (Zalaegerszeg, Zrínyi Miklós Gimnázium); *Ekler Márton* (Zalaegerszeg, Zrínyi Miklós Gimnázium).

Ízelítő a válogatóverseny feladataiból:.

Az elmúlt 200 évben Európa 2 országában feljegyezték, hogy a választások után milyen párt (baloldali vagy jobboldali) adta a miniszterelnököt. Olyan időszakokat keresünk, amikor az összes országban ugyanolyan jellegű párt volt hatalmon. A győztes adja a miniszterelnököt a választás napjától a következő választást megelőző napig.

Készíts programot (PARTOK.PAS vagy PARTOK.C), amely megadja azokat az időintervallumokat a legelső választás és az aktuális idő között, amikor mindegyik országban azonos jellegű párt volt hatalmon!

A PARTOK.BE állomány első sorában az aktuális dátum szerepel (év, hónap, nap, egy-egy szóközzel elválasztva. A második sorban az első országbeli választások száma van ($1\leqslant N\leqslant 200$), a következő N sor pedig egy-egy választás eredményét tartalmazza. Utána következik a második országbeli választások száma ($1\leqslant M\leqslant 200$), majd M sorban újra egy-egy választás eredménye. A választási eredmény első karaktere a B vagy a J betű, attól függően, hogy baloldali, vagy pedig jobboldali párt adta a miniszterelnököt, ezt követi három egész szám, egymástól és az első betűtől egy-egy szóközzel elválasztva: a választás éve, hónapja és napja.

A PARTOK.KI állomány első sorába azon időintervallumok K számát kell írni, amikor mindegyik országban azonos jellegű párt adta a miniszterelnököt, a következő K sorba pedig növekvő időrendben az egyes intervallumokat. Minden intervallum 6 egész számot tartalmaz egymástól egy-egy szóközzel elválasztva: az időszak elejét (év, hónap, nap) és végét (év, hónap, nap). Ha két egymáshoz illeszkedő intervallumban ugyanaz a párt adta mindkét országban a miniszterelnököt, akkor az egy (!) intervallumnak számít!

❄

A Nemes Tihamér OKSzTV III. kategóriájának 1–22. helyezettje számára olimpiai válogatóversenyt rendeztünk, amelynek alapján az idei Nemzetközi Informatikai Diákolimpián, Tamperében (Finnország) az alábbi tanulók indultak:

Novák Ádám (Eger, Neumann János KgSzKI és Gimnázium); *Ritter Ádám* (Fazekas Mihály Fővárosi Gyakorló Gimnázium); *Hargitai Gábor* (Ócsa, Bolyai János Gimnázium); *Földényi Tamás* (Fazekas Mihály Fővárosi Gyakorló Gimnázium).

Ízelítő a válogatóverseny feladataiból:.

A ProtoNet számítógépes hálózat úgy alakult ki, hogy eredetileg különálló hálózatokat összekapcsoltak. Mindegyik hálózat saját, egyedi protokollt használt. Az egyes részhálózatok az összekapcsolás után is a régi módon, azaz saját protokollt használva működnek. A hálózati hardvert azonban felszerelték olyan szoftverrel, amely képes bármelyik két protokoll közötti váltásra. A teljes hálózat most úgy működik, hogy ha egy X csomópontból közvetlenül egy Y csomópontba kell csomagot küldeni, és X, valamint Y nem azonos részhálózathoz tartozik, akkor előbb protokollváltást kell végrehajtani. A hálózati működést optimalizálni szeretnék. Ez azt jelenti, hogy olyan szoftvert kell készíteni, amely meghatározza, hogy ha egy adott A csomópontból egy másik B csomópontba kell küldeni a csomagot, akkor milyen útvonalat kell választani ahhoz, hogy a protokollváltások száma minimális legyen.

Készíts programot (PROTO.PAS vagy PROTO.C), amely

- Kiszámítja, hogy az A csomópontból a B csomópontba küldendő csomag esetén legkevesebb hány protokollváltás szükséges!
- Megad egy olyan A-ból B-be vezető útvonalat, amelyen a protokollváltás a lehető legkevesebb!

Eredményeink a 2001. évi Nemzetközi Informatikai Diákolimpián

2001. július 14-e és 21-e között rendezték meg a 2001. évi Nemzetközi Informatikai Diákolimpiát (IOI) Finnországban, Tamperében. A versenyen 72 ország 272 versenyzője vett részt.

Versenyzőink közül
Ritter Ádám (Fazekas M. Főv. Gyak. Gimn.),
Novák Ádám (Eger, Neumann János Szki.)
Földényi Tamás (Fazekas M. Főv. Gyak. Gimn.) ezüstérmet szerzett,
Hargitai Gábor (Ócsa, Bolyai János Gimn.) 5 ponttal maradt le a bronzéremről.

Ezek alapján csapatunk a nemhivatalos versenyben a 15. helyen végzett. A verseny az elmúlt éveknek megfelelően szakmailag erős, reális volt.

Zsakó László

Szoftver-kalauz

Derive

A *Derive* egy olyan matematikai program, amely segítséget nyújt számunkra az algebrai változókkal, kifejezésekkel, egyenletekkel, függvényekkel, vektorokkal és mátrixokkal végzett műveletek, eljárások végrehajtásában. A program legfőbb erőssége a szimbolikus algebra és a grafika, azaz a 2 és 3 dimenziós függvényábrázolás. Kiváló eszköz numerikus és szimbolikus matematikai számításokhoz, matematikai munkák dokumentálásához, valamint a matematika tanításához és tanulásához.

A diákok és a tanárok számára a Derive ideális segédeszköz. A program numerikus, algebrai és grafikus adottságai a matematika tanításának, tanulásának és megértésének újfajta megközelítését teszik lehetővé. Több témakörben eredményesebben és hatásosabban alkalmazható, mint a hagyományos megközelítés. Olyan vizsgálatok, amelyek technikai részletei alapos és fáradságos gyakorlást követelnek meg az iskolai gyakorlatban - például függvényábrázolás – a Derive segítségével most egy billentyű lenyomásával megoldhatók, elrejtve előlünk a hosszadalmas számítási eljárások részleteit. Amíg a Derive elvégzi a probléma megoldásának mechanikus ill. algoritmikus részét, addig a felhasználók a fogalmak matematikai jelentésére, összefüggéseire koncentrálhatnak. A technikai készségek sulykolása helyett a tanárok és tanulók a problémamegoldás izgalmas és hasznos módszereire összpontosíthatnak.

A KöMaL szeptemberi számában jelent meg a **C. 636.** gyakorlat:

C. 636. Ábrázoljuk a derékszögű koordinátarendszerben azokat a pontokat, amelyek koordinátáira

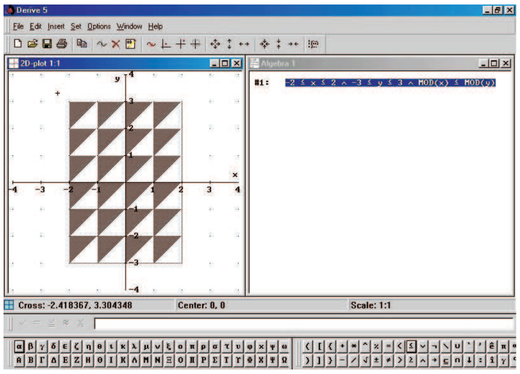
$$-2 \leqslant x \leqslant 2, \qquad -3 \leqslant y \leqslant 3, \qquad \{x\} \leqslant \{y\}.$$

(A $\{z\}$ a z szám törtrészét jelöli.)

A Derive segítségével könnyen és gyorsan megoldhatjuk ezt a példát. A program elindítása után, kiválasztjuk a menüből a 2-dimenziós ábrázolást. A szerkesztő-sorban a következő módon adhatjuk meg a feltételeket:

$$(-2 \leqslant x \leqslant 2) \wedge (3 \leqslant y \leqslant 3) \wedge \operatorname{mod}(x) \leqslant \operatorname{mod}(y).$$

A $\wedge$ jel – logikai „és” – felhasználásával írhatjuk fel, hogy mindhárom tulajdonságnak teljesülnie kell. Ezután már csak annyi dolgunk maradt, hogy kirajzoltassuk a programmal a pontokat. Eredményül a következő ábrát kapjuk.



A Windows környezetben futó angol nyelvű program installált állapotban kb. 4 Mbytenyi helyet foglal el a winchesteren. A program képernyőképe a következő komponensekből áll: címsor, menüsor, parancsikonok, „algebra ablak”, státuszsor, szerkesztő sor (a kifejezések beviteléhez), görög és matematikai szimbólumok eszköztára. Az interaktív helprendszer most is a rendelkezésünkre áll.

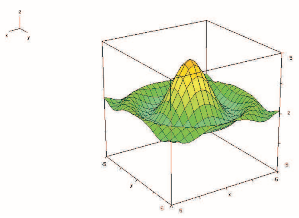
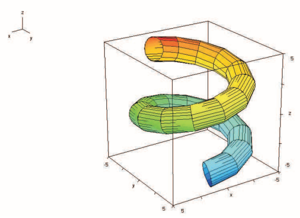
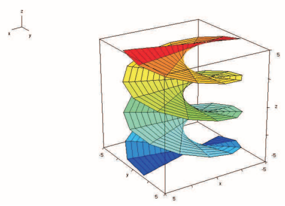
A Derive két matematikai témakörrel foglalkozik: algebra és analízis. Rövid bemutatásuk során nem törekszünk a teljességre. Azokat a lehetőségeket írjuk le, amelyeket közvetlenül a menük használatával valósíthatunk meg. Tudnunk kell viszont, hogy ezek csak egy töredékét adják a programban használható parancsoknak – ezek valójában függvények – amelyeket a szerkesztő sorba beírva aktivizálhatunk. Teljes listájukat a help tartalmazza, amit érdemes tanulmányozni, hiszen ismeretük nélkül a rendelkezésre álló segédeszközöknek csak egy részét hasznosítjuk.

Algebra

Foglalkozhatunk algebrai kifejezések különböző átalakításaival, tulajdonságaik vizsgálataival. Megoldhatunk egyenleteket, egyenletrendszereket, kiválasztva a megoldások alaphalmazát illetve hogy az eredményt numerikus vagy szimbolikus formában kérjük-e. Eközben lehetőségünk van a változók, részkifejezések helyettesítésére is. Egyszerűen megvalósíthatjuk a vektor és mátrix műveleteket, jellemző adataikat közvetlen parancsok megadásával kérhetjük a programtól.

Analízis

A témakörben megtalálható a határérték-számítás, a differenciál- és integrálszámítás, végtelen sorok és szorzatok kiértékelése. Lehetőség van magasabb fokú, illetve többváltozós deriváltak kiszámítására, függvények Taylor-sorba fejtésére, határozott és határozatlan, akár többváltozós integrálok előállítására is. A program komplex változós függvényeket is kezel. Differenciálegyenletek megoldása is lehetséges, bár ehhez már a helpre van szükségünk. Természetesen grafikonokat is tudunk készíteni. A Derive 2- és 3-dimenziós függvényábrázolása rengeteg lehetőséget nyújt. Meghatározhatjuk a grafikonok színét, nézetét, a nagyítás (kicsinyítés) mértékét. Az ábrákat pozícionálhatjuk, forgathatjuk, valamint animálásra is lehetőségünk van. Mivel a program alapvetően 3 ablakban dolgozik (algebra, 2D, 3D), ezért a grafikonokat képernyő nagyságnyi területeken csodálhatjuk. Egy ábrára több függvényt is kirajzoltathatunk, valamint ezeket különböző képformátumokba menthetjük el, illetve az algebra ablakba szemléltetésnek be is illeszthetjük.

A program rengeteg lehetőséget rejt magában. A középiskolai és felsőoktatási képzésben kiválóan hasznosítható. Komoly matematikai tudásbázisa van, de ennek kihasználásához természetesen elengedhetetlen a Derive használatának tökéletes ismerete.

A program 30 napos free változata letölthető a `www.derive.com` honlapról.

Számítástechnikából kitűzött feladatok (7–9.)

I. 7. A Faktoriális (N) függvény rendkívül gyorsan növekszik. Míg az $5! = 120$, addig már a $10! = 3\,628\,800$ ábrázolásához 4 byte-os egész számokra van szükség. A 100! pedig már csak speciális matematikai programokkal kezelhető.

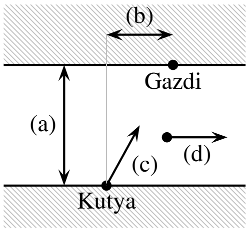
Tudjuk azonban, hogy minden természetes számnak elkészíthető a prímtényezős felbontása. Például:

$$5! = 2^3 \cdot 3 \cdot 5 \qquad 10! = 2^8 \cdot 3^4 \cdot 5^2 \cdot 7.$$

Készítsünk programot, amely beolvassa billentyűzetről N értékét ($1 \leqslant N \leqslant 10\,000$), majd kiírja a képernyőre az $N!$ prímtényezős felbontását.

(*10 pont*)

I. 8. Egy kutya úgy úszik át a folyón a túlparton álló gazdájához, hogy minden pillanatban a gazdi irányába igyekszik. Ezt a mozgást kell közelítő módszerrel modellezni, és a képernyőre kirajzolnod. Ehhez a következő, valós értékű paramétereket kell beolvasnia a programnak a billentyűzetről:



(a) A folyó szélességét méterben.

(b) A gazda távolságát méterben a kutya kezdőpontjának vetületétől a túlparton (pozitív, ha a folyásiránnyal azonos irányban van, negatív az ellenkező esetben).

(c) A kutya sebességét (m/s, nagysága állandó).

(d) A folyó sebességét (m/s, a folyó minden pontjában állandó nagyságú).

(e) A közelítés pontosságát, azaz annak az időintervallumnak a hosszát másodpercekben, amelyen belül a program egyenes vonalú mozgással számolhat.

A folyó két partját párhuzamos egyeneseknek tekintjük. A modellezés akkor álljon le, amikor a kutya már egy méternél közelebb kerül a túlsó parthoz.

Rajzoljuk ki a két partot jelző vízszintes egyeneseket, a kutya és a gazda kezdőhelyzetét és a kutya útját.

(*10 pont*)

I. 9. A binomiális együtthatók szokásos elrendezése (Pascal háromszög) az ábrán látható alakú lehet. A szélső elemek kivételével mindegyikre igaz, hogy a fölötte levő és az attól eggyel balra levő elem összege.

1					$N=6$	
1	1					
1	2	1				
1	3	3	1			
1	4	6	4	1		
1	5	10	10	5	1	
1	6	15	20	15	6	1

Készítsünk Excel táblázatot, amely ilyen módon képes megadni a Pascal háromszög első $N+1$ sorát! Az első sor hetedik cellájába lehessen beírni N értékét ($1 \leqslant N \leqslant 20$), és a táblázat minden esetben pontosan $N+1$ sorból álljon!

(*10 pont*)

A feladatok megoldásai a következő címre küldendők:

Cím: szamtech@komal.elte.hu

A beküldési határidő: **2001. december 13.**