

Programozás C és Matlab nyelven

Példatár C Zh-ra

1 Instrukciók

Jelen példatár a „Programozás C és Matlab nyelven” tárgy ZH felkészülését támogatja. A mellékelt könyvtárakban a Példák megoldása szerepel. Javasoljuk, hogy először saját erőből próbálják megoldani a Példákat, és csak ezek után nézzék meg az általunk mellékelt megoldásokat. A mellékelt megoldások nem a lelegegánsabb, hanem (talán) a legérthetőbb módon lettek megvalósítva.

2 Példák

2.1 Példa

- Írjon **feltolt** nevű függvényt, amely paraméterében átvesz egy 10 elemű int tömböt (**tomb** néven), majd sorsol 10 véletlen számot [0-9], és a **tomb** változóba statisztikát készít az egyes számok eloszlásáról (melyik számból mennyi van).
- Írjon **kiir** nevű függvényt, amely átvesz egy 10 elemű tömböt (**tomb** néven), és az elemeit kilistázza a képernyőre a következő formában: „0:10, 1:19 stb...”
- Írjon programot a fenti eljárások használatára.
- *Ötösért, ha minden más működik:* Írjon **ment** nevű függvényt, amely átvesz egy ilyen tömböt, és tartalmát egy **szamok.txt** nevű szöveges fájlba írja.

2.2 Példa

- Írjon **feltolt** nevű függvényt, amely paraméterében átvesz egy 13 elemű int tömböt (**tomb** néven), majd két dobókocka-dobás összegét (2d6) sorsolja ki 1000-szer, és a **tomb** változóba statisztikát készít az egyes eredmények eloszlásáról (a **tomb** indexe jelzi az összeget).
- Írjon **kiir** nevű függvényt, amely átvesz egy (**tomb** néven), és az elemeit kilistázza a képernyőre a következő formában: „0:0 \n 1:0 \n 2:2 stb...”
- Írjon programot a fenti eljárások használatára.
- *Ötösért, ha minden más működik:* Írjon **ment** nevű függvényt, amely átvesz egy ilyen tömböt, és tartalmát egy **szamok.dat** nevű int elemeket tartalmazó bináris fájlba írja.

2.3 Példa

- Hozzon létre **Tpont** nevű címkézett struct típust, amely egy pont egész típusú koordinátáit tárolja (x,y).
- Írjon **feltolt** nevű függvényt, amely átvesz egy file nevet, egy **Tpont** típusú fájlt létrehoz, és feltölti 20 db véletlen koordinátájú (-10,+10) pontokkal. A függvény visszatérési értéke legyen 0 sikeres megnyitás esetén, 1 sikertelenség esetén.
- Írjon **tavolsag** nevű függvényt, amely átvesz egy file nevet, bekér egy távolságot, és kilistázza a fájl azon pontjait, amelyek a (0,0) ponttól az adott távolságon belül vannak.
- Írjon programot a fenti eljárások használatára.
- *Ötösért, ha minden más működik:* Írjon **legkozelebb** nevű függvényt, amely felolvassa a fájlból a pontokat, és visszaadja azt a pontot, amely a (10,10) ponthoz a legközelebb van.

2.4 Példa

- Hozzon létre **TKosar** nevű struktúrát, amely egy kosárban lévő gyümölcsök számát tartalmazza (alma, korte, banan)
- Írjon **feltolt** nevű függvényt, amely egy TKosar binárisfájlt létrehoz, és feltölti 45 db véletlen TKosar elemmel (gyümölcsök száma 0-10)
- Írjon **legtobb** nevű függvényt, amely bekér egy számot (0-alma, 1-korte, 2-banan) és kilistázza azokat a kosarakat a fájlból, amelyekben a választott gyümölcsből van a legtöbb.
- *Ötösért, ha minden más működik:* Írjon **keres** nevű függvényt, amely kilistázza a fájl azon kosarait, amelyekben bármelyik két gyümölcs számának összege megegyezik a harmadik gyümölcs számával.

2.5 Példa

- Hozzon létre **TKor** nevű struktúra típust, amely egy kör egész típusú adatait tárolja (X,Y,R).
- Írjon **feltolt** nevű eljárást, amely egy bináris fájlt létrehoz, és feltölti 20 db véletlen középpontú (-10 +10) és véletlen sugarú (0-10) körrel.
- Írjon **origo** nevű eljárást, amely kilistázza a fájl azon köröit, amelyek lefedik a (0,0) pontot.
- *Ötösért, ha minden más működik:* Írjon **metsz** nevű eljárást, amely felolvassa a fájlból a köröket, és visszaadja azokat a köröket, amelyek nem metszik egyik koordinátatengelyt sem.

2.6 Példa

- Hozzon létre **TTeglap** címkézet struktúrát, amely egy téglalap két ellentétes csúcsának (egész) koordinátáit tárolja (x1,y1,x2,y2).
- Írjon **feltolt** nevű függvényt, amely egy bináris fájlt létrehoz, és feltölti 20 db véletlen koordinátájú (-5 .. +5) téglalappal.
- Írjon **terulet** nevű függvényt, amely kilistázza a fájl azon téglalapjait, amelyek területe nagyobb egy bekért számnál.
- *Ötösért, ha minden más működik:* Írjon **origo** nevű függvényt, amely kilistázza a fájl azon téglalapjait, amelyek lefedik a (0,0) pontot.

2.7 Példa

- Írjon kő-papír-olló játékot!
- Írjon **egykor** függvényt, amely átveszi a játékos pénzét a kör előtt, és visszaadja a kör után. A függvény a játék egy körét szimulálja: A játékos választhat egy jelet (0-kő,1-papír,2-olló), amely mellé a számítógép is sorsol egyet. Amennyiben a játékos nyer, a pénze nő 10-el, ellenkező esetben csökken ugyanennyivel.
- Írjon programot, ahol a játékos kezdeti pénze 100 Ft, és játékot játszik, amíg a játékos elveszti, vagy megduplázza a kezdeti pénzét.
- *Ötösért, ha minden más működik:* Írjon filehozfuz nevű függvényt, amelyet az egykor függvényből hívhat meg, és amely minden körben egy szöveges filehoz hozzáfűzi a két választást, és a játékos pénzét az alábbi formában: 'Jatekos-Ko, Gep-Ollo, penz: 90 Ft'

2.8 Példa

- Írjon szerencsejáték programot!
- Írjon **egykor** nevű függvényt, átveszi a játékos pénzét a kör előtt, és visszaadja a kör után. A függvény sorsoljon 5 számot [0..15] intervallumban (ne írja ki), majd kérdezze meg a játékost, hogy a számok között van-e két azonos? Amennyiben a játékos eltalálja, a pénze nő 5-el, ellenkező esetben csökken ugyanennyivel. (Ekkor már kiírhatja a számokat)
- Írjon programot, ahol a játékos kezdeti pénze 100 Ft, és játékot játszik, amíg a játékos elveszti, vagy megduplázza a kezdeti pénzét.
- *Ötösért, ha minden más működik:* Írjon **filehozfuz** nevű függvényt, amelyet az egykor függvényből hívhat meg, és amely minden körben egy szöveges filehoz hozzáfűzi a két választást, és a játékos pénzét az alábbi formában: 'Szamok: 1 2 15 9 0, penz: 95 Ft'

2.9 Példa

- Írjon **feltolt** nevű függvényt, amely paraméterében átvesz egy float típusú tömböt, illetve annak méretét, majd bekér két számot (a,b) és a tömb elemeit feltölti az $f_i = a + b * i$ számtani sor elemeivel, majd átadja azt a programnak.
- Írjon **osszeg** nevű függvényt, amely paraméterében átvesz egy float típusú tömböt, illetve annak méretét, bekér két egész számot (min, max), és a tömb *min* és *max* közti elemeinek összegét visszaadja a programnak.
- Írjon két eljárást, (**ment**, **listaz**) amely a tömböt file-ba menti, illetve file-ból felolvasva kilistázza.
- Írjon programot az eljárások alkalmazására, amely egy fix 100 elemű tömbbel dolgozik.
- *Ötösért, ha minden más működik:* tömb méretét a program kezdetekor kérdezze meg, és annak megfelelő méretű tömböt hozzon létre.

2.10 Példa

- Írjon csoki-automata szimulátor programot: Az automatában háromféle csoki van: (kerek 100 Ft, szögletes 120Ft, és lyukas 60 Ft) Kezdetben minden csokiból 10 db legyen az automatában, az adatokat program egy struktúra típusú tömbben tároljon.
- Írjon menüvezérelt programot az alábbi lehetőségekkel:
 - (1) Pénz bedobása (a felhasználó megmondja, hogy mennyi pénzt dob be, a bedobott pénzekhez ez hozzáadódik)
 - (2) Csokik vásárlása (Amennyiben nem elég a pénz, vagy elfogyott az adott csoki, közölje a felhasználóval. Ellenkező esetben adja ki a csokit, és írja ki a visszajáró összegét!)
 - (3) Kilépés
- Az egyes funkciókat (1,2) külön függvénnyel valósítsa meg!

2.11 Példa

- Írjon programot, amely kikérdezi az alapvető matematikai műveleteket
- Írjon **kerdez** nevű, int visszatérési értékű függvényt, amely sorsol egy feladatot (két véletlen számmal (1-10) véletlen egészértékű művelet (pl.: $9/2=4$) majd bekéri a felhasználó választát, és amennyiben a felhasználó megfelelően válaszolt 1, ellenkező esetben 0 értékkel tér vissza.
- Írjon programot, amely 10 feladatból álló tesztet készít a felhasználóval, majd közli az érdemjegyt: (0-6 találat:1, 7:2,8:3,9:4,10:5)
- *Ötösért:* a program minden feladatot, és az eredményt írja ki egy szöveges file-ba.

2.12 Példa

- Írjon torpedó programot! A tábla egy 5*5-ös karakter tömb legyen, amely az alábbi karaktereket tartalmazhatja: (O-semmi, H-hajó, S süllyedt, X nem talált.)
- Írjon **createtabla** nevű eljárást, amely a tömböt feltölti három hajóval.
- Írjon eljárást, amely bekér a felhasználótól két koordinátát, és találat esetén a tömb megfelelő helyére 'S', egyébként 'X' karaktert tesz.
- Írjon programot, amely a felhasználóval játékot játszik, amíg az összes hajó el nem süllyed
- **Ötösért:** Írjon **kiir** nevű függvényt, amely kirajzolja a táblát (A nem eltalált hajók helyére is 0 karaktert írva)

2.13 Példa

- Írjon egyszerű raktárkészlet-nyilvántartó programot:
- Hozzon létre **TAru** címkéjű struktúrát az alábbi mezőkkel: Név, Ár, Darab
- Írjon **feltolt** nevű függvényt, amely létrehoz egy aruk.dat nevű file-t, és feltölti 5db véletlen áruval (az áruk neve lehet egy véletlen karakter).
- Írjon **kiir** nevű függvényt, amely kilistázza a file-t.
- Írjon **kiad** nevű függvényt, amely bekéri az áru nevét, és a kiadandó mennyiséget, és ezt a fileban frissíti.
- Írjon **keves** nevű függvényt, amely kilistázza a beszerzendő árukat (amelyek készlete 5db alatt van)
- Írjon programot, amely a fenti funkciókat egy menüből kiválaszthatóan megvalósítja.

2.14 Példa

- Írjon programot, amely egy baromfiudvar életét szimulálja! Hozzon létre egy **Tbaromfi** nevű struktúra típust az alábbi mezőkkel: (tyúkok, tojások, hús darabszáma)
- Írjon **egynap** nevű függvényt, amely átvesz egy **Tbaromfi** típust, amely az állapotot írja le, és visszaadja az állapotot a nap végén az alábbi logika szerint:
 - Megkérdezi a felhasználót, hogy hány tojást, illetve húst akar eladni, illetve hány tyúkot akar levágni.
 - Először eladja a készleteket, majd levágja a baromfikat.
 - A régi tyúkok tojnak tojást, és a tegnapi tojásokból tyúk lesz.
- Írjon programot, amely kezdetben feltölt egy globális baromfiudvar változót 10-10-10db-al, majd ciklikusan megkérdezi a felhasználót, hogy aznap mennyit adott el belőlük.
- **Ötösért:** A program minden nap tevékenységét szöveges file-ban tárolja.

2.15 Példa

- Írjon **feltolt** nevű függvényt, amely véletlen egész számokkal feltölt egy paraméterben átvett 10 elemű int tömböt.
- Írjon **atlag** nevű függvényt, amely kiszámolja a paraméterében átvett 10 elemű int tömb elemeinek számtani középértékét és float típusként visszaadja.
- Írjon **ment** nevű függvényt, amely átvesz egy file nevet, egy 10 elemű int tömböt, és a tömb tartalmát, illetve az átlagot bináris file-ba írja, mindegyiket a saját típusa szerint. (10 int, 1 float)
- Írjon **listaz** nevű függvényt, amely az átvett filenév alapján a file elemeit a képernyőre listázza. Figyeljen a típusokra kiolvasásnál!
- Írjon programot az eljárások alkalmazására!

2.16 Példa

- Írjon eljárást, amely bekér egy egész számot, és egy ekkora méretű globális dinamikus tömböt feltölt véletlenszerű, (0-99) számokkal.
- Írjon eljárást, amely egy [0..9] méretű integer tömbbe statisztikát készít a számok eloszlásáról a tízes helyiértéken szereplő számjegyük alapján, majd az eredményt kiírja a képernyőre, és szöveges file-ba menti.

2.17 Példa

Írjon kóla automata programot!

Az automatában kétféle ital legyen: kóla (200Ft) és ásványvíz (150Ft). Mind a két italból 10 darab legyen az automatában. Az automata állapotát egy globális record típusban tárolja, amelynek mezői: kolak(kólák száma), vizek (vizek száma), bedobott pénz.

- Írjon menüvezérelt programot az alábbi lehetőségekkel:
 1. Pénz bedobása (a felhasználó megmondja, hogy mennyi pénzt dob be, a bedobott pénzekhez ez hozzáadódik)
 2. Kóla vásárlása
 3. Víz vásárlása
- Amennyiben nem elég a pénz, vagy elfogyott az ital, közölje a felhasználóval. Ellenkező esetben adja ki az italt, és írja ki a visszajáró összegét!
- Ötösért: Minden tevékenységről mentsen információt szöveges file-ba.

2.18 Példa

- Írjon **space** nevű függvényt, amely paraméterben átvesz egy sornyi szöveget (max. 200 karakter), és visszaadja az abban előforduló szóközök számát.
- Írjon programot, amely a **main.c** állományt soronként felolvassa, és a függvény felhasználásával kiírja a szóközök százalékos arányát (az összes karakterhez viszonyítva).

2.19 Példa

- Írjon **osztok** nevű függvényt, amely paraméterében átvesz egy egész számot, visszatérési értéként visszaadja az összes prímosztóját egy megfelelő (előre nem tudható) méretű int mutatóval (tömbbel). Az osztók számát paraméterként adja vissza.
- Írjon **kiir** nevű függvényt, amely átvesz egy számot, és az osztok nevű függvényt felhasználva kiírja az osztóit.
- Ötösért: Írjon **ment** nevű függvényt, amely a ugyanazt csinálja, mint a kiir, csak az eredményt szöveges file-ba is elmenti.
- Írjon programot az eljárások tesztelésére.