Ada I. géptermi zárthelyi
2000. november 3.
A csoport

Sablont tartalmazó feladattal kell kezdeni, utána lehet bármi!
A jegy: az elért pontok összege.

1. feladat: 1 pont
Készítsen sablon eljárást logaritmikus keresésre.
A sablon paraméterei: Az elemek típusa: tetszőleges típus; az indextípus: tetszőleges diszkrét típus; a vektor típusa: az elemeket rendezetten tartalmazó vektor, amely a megadott indextípussal indexelt; valamint a rendezési reláció: az elemtípuson értelmezett rendezés. Itt lehessen az alapértelmezett rendezés használata esetén a megadását elhagyni.

A megírandó eljárás egy vektort és egy keresendő elemet kap paraméterül, válaszként pedig egy logikai értéket és egy indexet ad vissza. A logikai érték jelzi, hogy a keresett elemet megtalálta-e, és ha igen, az index megadja a keresett elem indexét.

Példányosítsa a fenti sablont egy, a hét napjaival indexelt vektortípussal, amelyben az elemek egész számok és teszteljék legalább öt különféle vektorral és keresett elemmel. A vektorok és a keresett elemek lehetnek a programba bedrótozottak.

2. feladat: 2 pont
Írjon egy sablon eljárást vektor rendezésére, amelynek paraméterei a vektor elem típusa, ami tetszőleges típus lehet; az index típus, ami tetszőleges diszkrét típus lehet; a vektor típusa; és az elemeken értelmezett " $<$ " reláció, amelyet az alapértelmezett reláció esetén nem kell megadni.

Példányosítsa a fenti sablont a következő feladatra: Adott n pont a síkban. Rendezzük a pontokat a ponton és az origón áthaladó egyenes és az x tengely által bezárt szög szerint növekvő sorrendbe. (Bezárt szögön az egyenes és az x tengely közötti pozitív forgásirányú szöget értjük; ha a szög ugyanaz, akkor a " $<$ " reláció fennáll.)

3. feladat: 2 pont
Készítsen sablont a Sor típusra. A sablon paramétere a sor elemeinek típusa. A Sort egyirányú ciklikus lista segítségével valósítsa meg. Műveletei: üres sor létrehozása, elem behelyezése a sorba, első elem kivétele, adott elem kivétele, sor kiírása a képernyőre.

A sablon felhasználásával szimulálja egy bolt pénztáránál álló fizetők sorát. Olvasson be egész számokat a billentyűzetről: egy pozitív szám egy új vevőt azonosít, aki beállt a sorba; negatív szám esetén az adott abszolút értékű vevő elhagyta a sort; 0 esetén egy vevő fizetett és távozik. A program akkor ér véget, ha kiürül a sor, és legalább egy vevő fizetett.

4. feladat: 2 pont
Szemléltesse a következő játék menetét:
Egy tetszőleges méretű négyzetes táblánk van, minden négyzetben egy lámpa. Ha a lámpát - ami egyben kapcsoló - megnyomjuk: addigi állapotának ellenkezőjére vált (elalszik vagy felgyullad) és ezt teszik (önmagukhoz képest) az eggyel felette, alatta, és mellette lévő lámpák is. (Amennyiben valamelyik hiányzik - tábla széle - a többi attól ugyanúgy változik.) A játék célja: a tábla összes lámpáját felgyújtani a teljesen kikapcsolt állapotból.
A tábla típust csomagban kell átlátszatlanul definiálni, de a típus paramétere kell legyen a tábla mérete (ami egyetlen érték, mivel a tábla úgyis négyzetes). A tábla műveletei: Inicializálás, Lámpa kapcsolása, Tábla kiírása. A főprogram feladata a megnyomandó lámpa koordinátáinak beolvasása, illetve kilépés "k" betű beolvasásakor. Minden lépés után rajzoljuk ki a tábla aktuális állapotát (a lámpákat pl. 0-val és 1-gyel ábrázolva). Ha az összes lámpát felkapcsoltuk, a program szintén terminál.

Ada I. géptermi zárthelyi
2000. november 3.
B csoport

Sablont tartalmazó feladattal kell kezdeni, utána lehet bármi!
A jegy: az elért pontok összege.

1. feladat:1 pont
Készítsen sablon eljárást a feltételes maximumkeresés programozási tételének megvalósítására. A sablon paraméterei: az elemtípus: tetszőleges típus; az index típus: tetszőleges diszkrét típus; a vektor típus: az elemeket tartalmazó vektor, amely a megadott indextípussal indexelt; a Béta tulajdonság, amely a keresés feltétele (az elemtípushoz logikai értékeket rendelő függvény); valamint egy rendezési reláció: az elemtípuson értelmezett rendezés. Itt lehessen az alapértelmezett rendezés használata esetén a megadását elhagyni.

A megírandó eljárás paraméterül egy vektort kap, amelyen a keresést végzi. Kimenő paramétere, hogy talált-e megfelelő tulajdonságú elemet, és ha igen, akkor a maximum indexe és értéke.

A sablon felhasználásával írjon programot, amely a billentyűzetről beolvassa egy egész számokat tartalmazó vektor méretét majd az elemeit és végül kiírja a legnagyobb negatív elemet a beolvasott számok közül.

2. feladat: 2 pont
Írjon egy sablon eljárást vektor rendezésére , amelynek paraméterei a vektor elem típusa; az index típus, ami tetszőleges diszkrét típus lehet; a vektor típusa; és az elemeken értelmezett " $>$ " reláció, amelyet az alapértelmezett reláció esetén nem kell megadni.

Példányosítsa a fenti sablont a következő feladatra: Adott n pont a síkban. Rendezzük a pontokat: a pont origótól számított távolsága szerint csökkenő sorrendbe. (Ha a távolság ugyanaz, akkor a " $>$ " reláció fennáll.)

3. feladat: 2 pont
Készítsen sablont a Sor típusra. A sablon paramétere a sor elemeinek típusa. A sort kétirányú aciklikus lista segítségével valósítsa meg. Műveletei: üres sor létrehozása, elem behelyezése a sorba, első elem kivétele, adott elem kivétele, sor kiírása a képernyőre.

A sablon felhasználásával szimulálja egy benzinkútnál tankoló autók sorát. Olvasson be egész számokat a billentyűzetről: egy pozitív szám egy új autót azonosít, aki beállt a sorba; negatív szám esetén az adott abszolút értékű autó elhagyta a sort; 0 esetén egy autó tankolt és távozik. A program akkor ér véget, ha kiürül a sor, és legalább egy autó tankolt.

4. feladat: 2 pont
Szemléltesse az amőba játék menetét:
Egy tetszőleges méretű négyzetes táblánk van, ezen amőbázhatunk üres táblából indulva. Mindig a kör kezd.

A tábla típust csomagban kell átlátszatlanul definiálni, de a típus paramétere kell legyen a tábla mérete (ami egyetlen érték, mivel a tábla úgyis négyzetes). A tábla műveletei: Inicializálás, Figura elhelyezése (csak a pozíció adott), Tábla kiírása. A főprogram feladata a következő játékos lépésének beolvasása, illetve kilépés "k" betű beolvasásakor. Egy lépés után mindig rajzoljuk ki a tábla aktuális állapotát. Ha a táblán már sehova sem lehet tenni, a program szintén terminál.