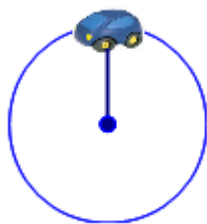


Regionális forduló

2012. november 24.

11-13. osztályosok feladata

8-13. osztályosok feladata



Egy játékgár az olimpiához kapcsolódva egy speciális, „mechanikus” reklámtáblát készít. Azt tervezik, hogy az olimpiai ötkarikán játékautókat mozgatnak körbe-körbe. Az autók mozgását körönként egy-egy speciális motor végzi, amelyeket a körök középpontjában helyeznek el. A motorokhoz erősített vékony rudak végén rögzítik a kisautókat, melyek így a köríven mozoghatnak. A kisautók színe megegyezik a karikák színével, minden körön

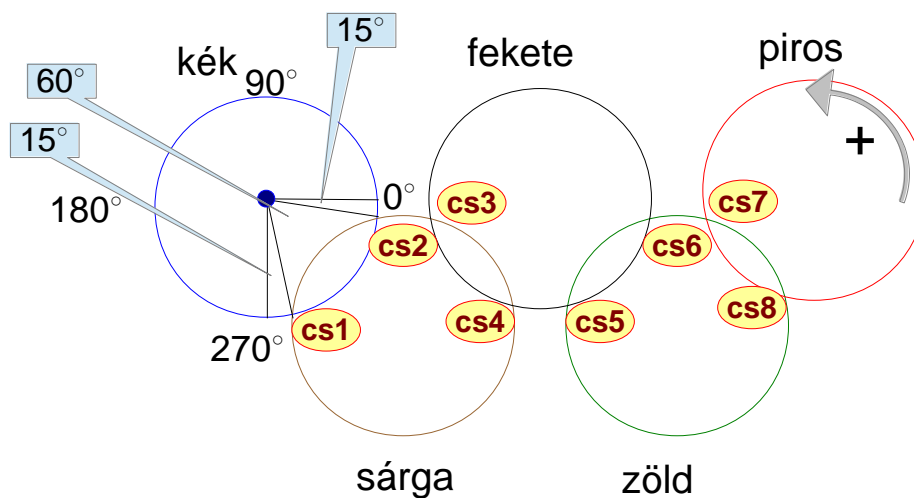
egy kisautó van.

Az autók mozgását számítógép segítségével szeretnék megtervezni, ezért egy szimulációs program elkészítésére kérnek titeket.

A gyártók sokféle mozgásformát ki akarnak próbálni, ezért a programnak sokféle problémát kell kezelnie: Egy adott pillanatban hol vannak a kisautók, mekkora a sebességük, lesz-e ütközés, és ha igen mikor, mennyi idő alatt tesznek meg egy teljes kört, stb.

Az egyes autók sebessége nem biztos, hogy egyforma lesz; többféle kezdőhelyről indulhatnak, nem feltétlenül egyszerre indulnak, nem mindegyik kering azonos irányban a körökön.

Ha két kör metszéspontjához egyszerre ér két kisautó, összeütköznek, és ez a szerkezet meghibásodását okozza. Az is probléma, ha két kisautó egy időben a másik kör területén belül van, ilyenkor a rudak összeakadhatnak.



Az ábrán a tervezett pálya, az olimpiai ötkarika látható. A karikák színe a feliratok alapján azonosítható. A feladat elkészítéséhez szükséges szögértékek leolvashatóak, a körök metszéspontjait, a „csomópontokat” sorban a cs1, cs2, ... cs8 jelölik.

A csomópontok pozícióját az egyes körökön az alábbi táblázat tartalmazza: (A megfelelő kör színe és a csomóponthoz rendelt pozíció fokban szerepel az adatokban.)

cs1	(kék:285,	sárga:165)
cs2	(kék:345,	sárga:105)
cs3	(sárga:75,	fekete:195)
cs4	(sárga:15,	fekete:255)
cs5	(fekete:285,	zöld:165)
cs6	(fekete:345,	zöld:105)
cs7	(zöld:75,	piros:195)
cs8	(zöld:15,	piros:255)

A program elkészítéséhez szükséges feltételeket a játékgyár

a következő módon adta meg:

- A kisautók a körökön 4 helyről indulhatnak, a 0, 90, 180 vagy 270 fokkal jelölt pozícióból. (Ezeket csak a kék körön jelölték be.)
- A kisautóknak háromféle sebességük lehet. Egy másodperc alatt a tengelyükhöz erősített rudat a motorok 1, 3 vagy 5 fokkal tudják elfordítani.
- A forgásirány pozitív, ha az óramutató járásával ellentétes, ellenkező esetben negatív.
- Nem minden körön mozog az autó. A mozgó autók nem feltétlenül egyszerre indulnak.
- Ha egy kisautó úgy ér egy csomóponthoz, hogy a körpályája belsejében a metsző körön egy másik autó fut, akkor el kell dönteni¹, hogy a hajtókarok összeakadása nélkül tovább tud-e haladni. Ha igen, akkor tovább halad, ha nem, akkor a adott ideig² várakoznia kell.
- Ha egy csomóponthoz két autó ér egyszerre, akkor ütközés történik, mindegyik autó megáll. A kísérletnek vége szakad.
- A kisautók méretétől eltekintünk. **Nem tekintjük ütközésnek** azt az esetet, amikor a **csomópontban várakozó** autó mellett a csomópontban elhalad egy másik, mozgó autó.

¹ A hajtókarok biztosan összeakadnak, ha a két közös csomópont között egymással szembe haladnak a kisautók. A hajtókarok összeakadhatnak, ha az autók haladási iránya megegyezik, és a csomóponthoz később érkező autó nagyobb sebességű, vagy ha a csomóponthoz később érkező autó sebessége nem nagyobb a másikénál, de az még nem tette meg a csomópontok közötti távolság felét (30 fokot).

² Ha a másik autó szembe jön, akkor addig kell várakozni, amíg az ki nem ér a közös területről. Ha a másik autó azonos irányba mozog, de lassabban halad, addig, amíg az ki nem ér a közös területről. Ha a másik autó azonos irányba mozog, és nem lassabb, akkor addig, amíg a másik autó mozgató karjának elfordulása a közös részen el nem éri a 30 fokot, meg nem teszi a közös részre eső útjának felét.

Bemeneti adatok

A gyártó azt kéri, hogy az általa tesztelni kívánt mozgásváltozatokat a program a bemeneti paraméterek alapján elemezze. Ezeket a kezdeti paramétereket a **kisautok.txt** nevű fájl tartalmazza.

A bemeneti fájl minimum 1, maximum 5 sorból áll, attól függően, hogy melyik körökön mozognak autók.

Egy sorban mindig 5 adat van egymástól egy szóközzel elválasztva. Ezek sorrendben a következők:

1. az autó színe (ékezetes betűk használata nélkül)
2. az autó sebessége (1, 3 vagy 5 fok/sec)
3. kezdőpozíció (0, 90, 180 vagy 270 fok)
4. menetirány (+ vagy -)
5. a késleltetés másodpercben az első autó indulásához képest

Pl.: piros 3 0 + 5
 kek 1 90 - 0
 fekete 5 180 + 3

A fájlban a színek sorrendje bármilyen lehet.

A program feladata

1. A program induláskor a felhasználó bevonása nélkül olvassa be az aktuális paramétereket a bemeneti fájlból, majd kérjen be egy időpontot, amelynek alapján számolja ki, és a **képernyőre írja ki** az alábbi adatokat (a bekért időpont az időmérés kezdetétől, a 0 időpillanattól eltelt, másodpercben mért idő):

- A kisutók helye: 0 vagy annál nagyobb, de 360 foknál kisebb szög az ábrán megadott módon
- A kisautók sebessége: 0 vagy az autó induláskor érvényes sebessége fok/másodpercben
- Az egyes kisautók által az adott időpontig megtett teljes körök száma. (Minden színre külön.)
- Az adott időpontig összesen mennyi ideig váraoztak az egyes autók. (Csak azoknak az autónak kell megadni külön-külön az összesített váraozási idejét, amelyek csomópontban váraoztak. Az el sem indulókat nem kell kiírni.)
- Az egyes kisautók mennyi idő alatt tették meg az első teljes kört. Ha ütközés miatt a szerkezet leállt, akkor csak azokat kell felsorolni, amelyek legalább egy teljes kört megtettek az ütközés időpontjáig. (A váraozás ideje természetesen növeli a teljes kör befutási idejét.)
- Ha az adott időszakban volt ütközés, akkor az ütköző autók színét és az ütközés idejét.
- A szerkezet által 10 perc alatt elhasznált energiát az 5 autóra összesítve.

Az autók indítása 8 egység energiát igényel, a leállítása 5 egységnyit.

A mozgathoz szükséges energia függ a sebességtől. Ha a motor másodpercenként egy fokkal fordít, akkor ez másodpercenként 2 egység energiát jelent, ha három fokkal, akkor ez 6

egységet, és az öt fokos forgatás 10 egységnyit.

A felhasználóbarát felület és az egyértelmű adatközlés érdekében az adatok kiírásánál mindig közölni kell, hogy a megjelenített adat mire vonatkozik. Az is fontos, hogy az adatokat tetszőleges ideig lehessen tanulmányozni. (Ez megoldható úgy, hogy az összes kiírt adat egyszerre látszódik a képernyőn, vagy görgetéssel/lapozással láthatóvá tehető bármelyik.)

2. Írja ki a **cspont.txt** fájlba, hogy a megadott időszakban az egyes csomópontokon melyik kisautó haladt keresztül és mikor! Ha az adott csomóponton nem haladt át autó, akkor ezt kell jelezni a „nem haladt át autó” szöveg kiírásával

A **cspont.txt** fájl pontosan 9 sorból áll. Az első sor a megadott időtartamot/időpontot tartalmazza, a további sorok pedig rendre az egyes csomópontokon áthaladó autókra vonatkozó adatokat az alábbiak szerint:

Minden sor a csomópont azonosítóval kezdődik, majd az áthaladás sorrendjében következnek az autókra vonatkozó információk; az áthaladás ideje és az áthaladó autó színe.

```
Pl.      180
         cs1 165:kek
         cs2 105:kek
         cs3 6:feke 78:feke 150:feke
         cs4 18:feke 90:feke 162:feke
         cs5 24:feke 96:feke 168:feke
         cs6 36:feke 108:feke 180:feke
         cs7 70:piros
         cs8 90:piros
```

A szóközök és az egyéb elválasztójelek használata nem kötelező, ezek tetszőlegesen alkalmazhatóak, de a fájl olvashatóságát biztosítaniuk kell.

Beadandó

A program **forráskódja** (a programozási környezettől függően a forráskód több fájl is lehet, esetleg mappák is tartozhatnak hozzá) és a lefordított **reklam.exe** állomány.

A bemeneti adatokat tartalmazó fájl mindig az aktuális (fejlesztői környezetben az alapértelmezett) könyvtárban található. A kimeneti fájlt is ide kell menteni. (A program kódja nem tartalmazhat abszolút elérési útvonalat.)

Feltételezzük, hogy a bemeneti fájlokban szereplő adatok minden esetben helyesek.

Kódolási alapelvek

A forráskód minőségét is értékeljük.

Irányelvek, szempontok:

- Egységes kódolási szabályok az azonosítókra:
 - a változók egységes elnevezése (kis- és nagybetűk vagy más speciális karakterek használata),
 - az osztályok egységes elnevezése (objektum-orientált programnyelv esetén, pl. C++),
 - a függvények és eljárások tartalomra utaló elnevezése,
 - a programkód egységes strukturáltsága, tagoltsága.
- A kód minősége (könnyen – emberek számára – érhető illetve karbantartható kód):
 - áttekinthető, lehetőség szerint rövid eljárások, függvények, fájlok,
 - beszédes, tömör elnevezésű azonosítók,
 - objektum-orientált nyelveknél globális változók mellőzése.
- Kommentezés:
 - A kommentezés elsődleges célja, hogy a programban a **miért?** kérdésre adjon választ. (A **mit?** kérdésre az azonosítók megfelelő elnevezése és a megfelelően strukturált kód, a **hogyan?** kérdésre pedig az áttekinthető forráskód ad választ.)
 - A túlzásba vitt kommentezés csökkenti az áttekinthetőséget, a túl kevés komment nehezíti a megértést.
 - **Elvárás a változók, osztályok, függvények és eljárások szerepének rövid, értelemszerű kommentezése.**

Elérhető pontszám: 120 pont (Ebből a helyes dokumentálás – kommentezés – 15 pont)

Jó munkát kíván a Versenybizottság!