

Regionális forduló

2010. november 27.

A 9-10. osztályosok feladata

Egy rendezvényszervező cég titeket kért meg arra, hogy a rendezvényeik megszervezését és lebonyolítását segítő programot készítetek számukra. A kész programot a projektmenedzser fogja használni, aki minden esetben (minden projektben) az alábbi adatok alapján dolgozik:

- a projekt során elvégzendő tevékenységek száma,
- az egyes tevékenységek jellemzői (a tevékenység egyedi azonosítója és neve, az időigénye napokban, a megelőző tevékenység(ek), az egyes tevékenységeket követő esetleges várakozási idő).

A ti feladatotok egy olyan program megírása, amelyik a kapott információk alapján megadja, hogy

- mennyi az a legrövidebb időtartam, ami alatt elvégezhető a projekt,
- az egyes tevékenységek elkezdésének mi a legkorábbi időpontja.

Készítenetek kell egy **sávós ütemtervet** is, amely a képernyőn jeleníti meg a feladatok időbeli eloszlását.

Bemeneti adatok

A projektet alkotó tevékenységek listáját a **tev.txt** fájl tartalmazza.

Az első sorban a tevékenységek számát tüntettük fel, a további sorok pedig ezen tevékenységek leírását adják.

- A sor elején a tevékenység egykarakteres azonosítóját kettőspont és szóköz követi, ezután vesszővel és az azt követő szóközzel elválasztva a tevékenység leírása (neve), időigénye napokban, a tevékenységet megelőző¹ korábbi tevékenység azonosítója következik. (A tevékenység neve nem tartalmaz vesszőt.)
- Ha egy tevékenységet nem kell megelőznie más tevékenységnek, akkor a tevékenység neve helyett a **nincs** szó szerepel.
- A tevékenységek időtartama legalább 1 nap. (Mindig egész szám.)
- Ha az aktuális tevékenységet több korábbi tevékenység előzi meg, akkor azokat **csak** szóköz választja el.
- Ha az aktuális tevékenységet megelőző korábbi tevékenység befejezése után várakozni kell, akkor a szükséges várakozási idő kerek zárójelben szerepel közvetlenül a tevékenység-azonosító után.

¹ Ha az X tevékenységet Y tevékenység előzi meg, az azt jelenti, hogy X mindaddig nem kezdhető el, ameddig Y be nem fejeződött.

Kimeneti adatok

A kimeneti adatokat részben a képernyőn kell megjelenítenetek, részben pedig kimeneti szövegfájlba kell kiírnotok.

A képernyőre készítenetek kell egy táblázatszerű sávos ütemtervet, amelynek a fejlécén a napok sorszámát kell feltüntetni, a további soraiban pedig az egyes tevékenységek azonosítója után vízszintes vonallal kell megjelölni azokat a napokat, amelyeken a tevékenységet végezni kell. Ha több napon át végeznek egy tevékenységet, akkor az érintett napokat folytonos vonallal is meg lehet jelölni.

Az alábbi két ábra egy-egy példát mutat egy tevékenységsor idősávjának lehetséges megjelenítésére. (Más vonalstílus is választható.)

	1.	2.	3.	4.
A		—		
B		—	—	
C			—	—

1. ábra

	1.	2.	3.	4.
A		—		
B		————		
C			————	

2. ábra

Az ütemtervet karakteres, illetve grafikus képernyőn egyaránt elkészíthetitek. Nem jelent előnyt a grafikus képernyő használata. (Az értékelésben csak az ábra helyességét és áttekinthetőségét értékeljük.)

A képernyő alsó soraiban lehetőséget kell biztosítani a felhasználónak arra, hogy a tevékenység kódjához a tevékenység nevét lekérdezhesse.

A program futása során **utem.txt** néven készüljön egy kimeneti fájl is!

- A fájl első sora a projekt elvégzéséhez minimálisan szükséges napok számát mutassa!
- A további sorok az egyes tevékenységek egyedi azonosítói után a tevékenység legkorábbi kezdési időpontját tartalmazzák! A tevékenység-azonosító után kettőspont és szóköz szerepeljen!
- A tevékenységek az azonosító szerinti betűrendben kövessék egymást!

Feladat

Készítsetek programot, amelyik a fent ismertetett formátumú tevékenységlista alapján a kért formátumban megadja a kimeneti adatokat!

Feltételezzük, hogy a bemenő adatok mindig olyanok, hogy létezik megfelelő ütemezés. Egy projekt maximálisan 20 tevékenységből áll, bármelyik projekt időtartama maximálisan 20 nap, és nem fordulhat elő, hogy 2 tevékenységnél többet lehessen párhuzamosan végezni.

Az alábbi példa egy lehetséges projekt megadott szempontok szerinti bemeneti és kimeneti adatait mutatja be.

Példa: versenyszervezés

Bemeneti adatok

A **tev.txt** fájl tartalma:

12

A: a versenykiírás elkészítése és a verseny meghirdetése, 5, nincs

B: a nevezések fogadása és összesítése, 3, A(1)

C: a versenyfeladat kidolgozása, 4, nincs

D: a javítókulcs kidolgozása, 2, C

E: a feladat és a javítókulcs ellenőrzése, 3, D

F: a nevező csapatok értesítése és tájékoztatása, 1, B

G: a feladatlapok előkészítése, 2, E

H: a helyszín előkészítése, 1, F G

I: a versenyzők fogadása és a verseny lebonyolítása, 1, H

J: a versenymunkák javítása, 4, I

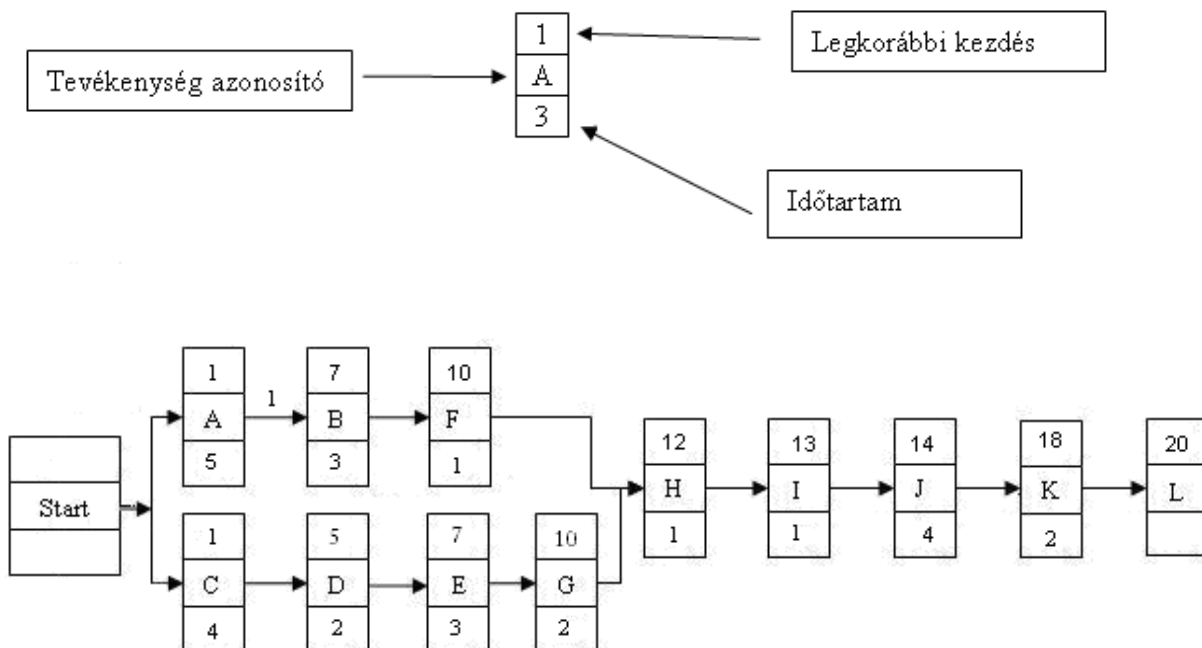
K: a javítás ellenőrzése, 2, J

L: eredményhirdetés, 1, K

Az adatok feldolgozása

Az alábbi ábra segítségével át tudjátok tekinteni a bemeneti és a kimeneti adatok közötti kapcsolatokat:

- a projekt lebonyolításához 20 napra van szükség,
- az A, B...L tevékenységek mikor kezdődhetnek.



Kimeneti adatok

A képernyőkép:

	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.	11.	12.	13.	14.	15.	16.	17.	18.	19.	20.		
A	_____																					
B							_____															
C	_____																					
D					_____																	
E							_____															
F										-												
G										_____												
H												-										
I													-									
J														_____								
K																			_____			
L																					-	

A tevékenységek nevét a kódjuk alapján lekérdezheti.

Adja meg a kódot!

C: a versenyfeladat kidolgozásaAz **utem.txt** tartalma:

20
A: 1
B: 7
C: 1
D: 5
E: 7
F: 10
G: 10
H: 12
I: 13
J: 14
K: 18
L: 20

Beadandó

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

A bemeneti adatokat tartalmazó fájlok mindig az aktuális (fejlesztői környezetben az alapértelmezett) könyvtárban találhatóak. A kimeneti fájlokat 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: 100 pont (Ebből a helyes dokumentálás – kommentezés – 15 pont)

Jó munkát kíván a Versenybizottság!