

XIII. ROCS – döntő

6. évfolyam – II. kategória

2022. április 23.

1. feladat (20 pont)

Írj egy olyan programot, amely a következő feladatot hajtja végre! A mellékelt állományban található számokat olvassa be és jelenítsen meg minden sorban annyi fekete színnel kitöltött négyzetet amennyi a beolvasott szám! A négyzet mérete tetszőleges lehet, de a képernyőre ki kell férnie legalább 5 négyzetnek egy sorban és legalább 5 sornak. A négyzetek és a sorok között legyen azonos méretű (nem 0) távolság!

Az állomány első sorában lévő szám azt mutatja meg, hogy hány számot kell az állományból beolvasni. A *touch* szenzor megnyomására ugyanezeket a négyzeteket oszlopokban jelenítse meg. A fájlban tárolt sorrendnek megfelelően.

A sorban, oszlopban való megjelenítést a *touch* szenzor megnyomásával lehessen váltani mindaddig, amíg a programot ki nem kapcsoljuk.

pl.:

állomány:

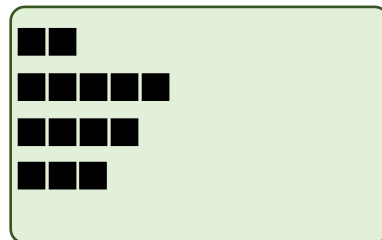
4 (ez azt jelenti, hogy az állományban a következő 4 sorban található számoknak megfelelő számú négyzetet kell megjeleníteni a robot képernyőjén)

2

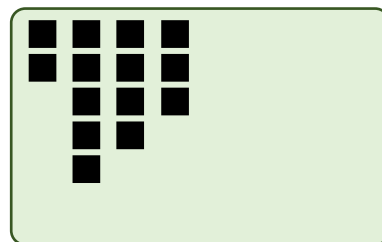
5

4

3



Touch szenzor megnyomása után (a sorok és oszlopok felcserélődtek, de a négyzetek száma nem változott)



A számokat a fájlból kell a csapatnak beolvasnia, nem írhatja be fixen a programba.

A fájl a csapat hozza létre. Minden szám külön sorban szerepel.

A fájl tartalma: 5 2 3 1 4 5

A programot mentsd el *feladatsorszám_csapatsorszám* néven! Pl.: ha csapat a 8-as számot húzta a regisztrációnál, akkor a fájl (projekt) neve: *1_8.ev3*

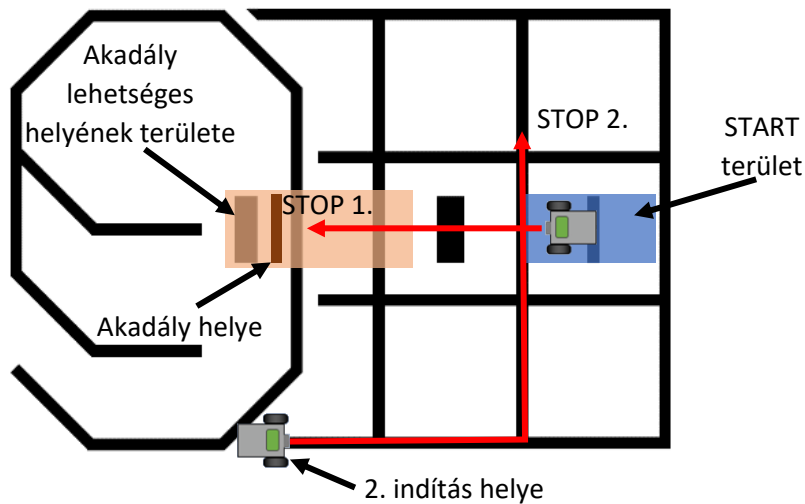
2. feladat (15 pont)

Írj programot, amely végrehajtása során a robot a következő feladatot hajtja végre! A robot az 1. indításkor elindul a startpozícióból egyenesen előre egy csíksor felett, addig megy amíg 10 cm-en belül akadályt érzékel és akkor megáll. A csíkok között biztosan van egy szélesebb és az is biztos, hogy az nem az első csík. (A lehetséges indítási pozíciókat a kék színű terület jelzi az ábrán.)

Ezután a robotot át kell helyezni a 2. indítási pozícióba, anélkül, hogy a programja leállna.

A robot csak a *touch* szenzor megnyomása után induljon el ismét a 2. pozícióból. A 2. pozícióból vonalkövetéssel kell haladnia és azután a vonal után, ahányadik a vastag vonal volt start pozícióból való indulás után, forduljon balra és folytassa a vonalkövetést. A robot a vonalkövetést addig folytassa amíg áthalad az annyiadik kereszteződés felett, ahányadik a vastag vonal volt.

pl.: Ha a 2. vonal volt a vastag vonal, akkor a robot a 2. start pozícióból indulva követi a vonalat a 2. elágazásig ott balra fordul majd ismét a 2. elágazásig követi a vonalat, ami után megáll.



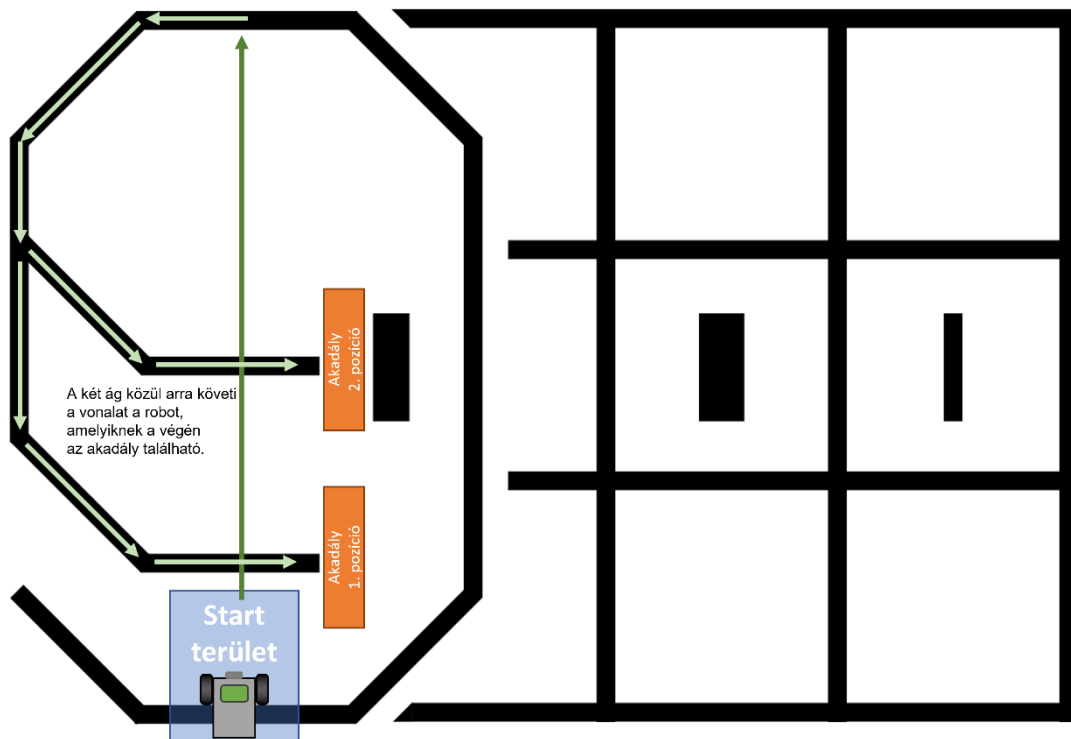
Az 1. indítás helye és az akadály pozíciója csak a bemutató során lesz ismert. Az akadály a barack színű területen lesz valahol.

A programot mentsd el *feladatsorszám_csapatsorszám* néven! Pl.: ha csapat a 8-as számot húzta a regisztrációnál, akkor a fájl (projekt) neve: *2_8.ev3*

3. feladat (17 pont)

Írj egy olyan programot, aminek a végrehajtása során a robot a következő feladatot hajtja végre! A robot a startpozícióból indul egyenesen előre. Jobbra a két fekete csík egyikének a végénél egy akadály található (akadály az 1. és 2. pozíció egyikében lesz). A 3. vonal után a robot forduljon balra és kezdje el követni a vonalat. Az elágazásnál arra menjen a robot, amelyik vonal végénél az akadály van és az akadály előtt 10 cm-rel álljon meg!

Mindkét akadálypozíció esetén be kell mutatni a megoldást. Az ábrán a kék színű terület jelöli a startpozíciót.



A programot mentsd el *feladatsorszám_csapatsorszám* néven! Pl.: ha csapat a 8-as számot húzta a regisztrációnál, akkor a fájl (projekt) neve: *3_8.ev3*

4. feladat (21 pont)

Írj programot, mely kiírja a képernyőre egy „majdnem” Fibonacci sorozat elemeit!

Sorsold ki a sorozat első elemét; ez legyen egy 1 és 4 közötti szám. Sorsolj egy másik számot, mely biztosan nagyobb, mint az első szám, és legfeljebb 6. Ez lesz a sorozat második eleme.

Sorsolj egy számot, mely megadja, hogy hány sorozatelem lesz kiírva; ez legyen egy 4 és 7 közötti szám.

Mindhárom sorsolt számot írd ki egymás alá a képernyőre, a sorsolás sorrendjében.

A negyedik sorba írd ki egymás mellé a sorozat elemeit, melynek képzési szabálya a következő: a harmadik elemtől kezdve minden elem egyenlő az előtte lévő két elem összegével.

A sorozatelemek kiírásánál használd a legkisebb betűméretet!

Például:

Az elsőnek sorsolt szám: 3

A másodiknak sorsolt szám: 5

7 db elem kiírása jött ki a sorsolásnál.

A sorozat elemei a képen az alsó sorban láthatók.

3
5
7
3 5 8 13 21 34 55

A programot mentsd el *feladatsorszám_csapatsorszám* néven! Pl.: ha csapat a 8-as számot húzta a regisztrációnál, akkor a fájl (projekt) neve: *4_8.ev3*

Spyke robot esetén a változtatás:

Minden képernyőn megjelenő szám ütközésérzékelő megnyomásáig látszódjon. A példánál:

A képernyőn megjelenik a 3. Ütközésérzékelő megnyomására az 5, újabb ütközésérzékelőre a 7.

A sorozat elemei ezután jelennek meg, mindegyik megjelenése egy ütközésérzékelő megnyomása után történik. Ha a szám kétjegyű, akkor számjegyenként jelenik meg: az első számjegy 3 másodpercig látszik, majd utána megjelenik a második számjegy.