1. feladat: Motorcsónak (34 pont)

Van egy programozható motorcsónakunk. A jármű kétféle parancsra hallgat: a **B** hatására balra, a **J** hatására pedig jobbra fordul 90 fokot. A tavon a nyilat (🡅) tartalmazó mezőről indul a nyíl által meghatározott irányba. Folyamatosan halad előre, és az irányán csak bójához érkezéskor módosít. Teli bójánál (⚫) beolvas egy parancsot, és az annak megfelelő irányba fordul. Csillagos bójánál (✪) azonban a legutóbb végrehajtott parancsot hajtja végre újra. Ha hullámos mezőn (🌫) halad át, akkor pedig a következő bóját figyelmen kívül hagyja (üres mezőnek tekinti). A csónak csak akkor áll meg, ha teli bójához érkezett, és elfogytak a parancsok. Ha a tópartnak ütközik, akkor azonnal elsüllyed, amit természetesen nem szeretnénk.

A tavat alkotó mezőkre az oszlopok feletti betűkkel és a sorok melletti számokkal tudunk hivatkozni.

Lássunk egy példát!

|  |  |
| --- | --- |
| A képen keresztrejtvény, fekete látható  Automatikusan generált leírás | A csónak a B4-es mezőről indul a nyíl irányába, és a **JB** kódot adtuk meg neki. Folyamatosan halad a B1-es mezőig, ahol beolvassa az első parancsot, és annak megfelelően jobbra fordul. A D1-es mezőhöz ér, ahol ismét jobbra fordul. Áthalad a D2-es mezőn, ezért a D4-es mezőn található bóját figyelmen kívül hadja. A D5-ös mezőre érve beolvassa a második parancsot, amelynek hatására balra fordul. Az E5-ös mezőn pedig megáll, mert nincsen több parancs. |

A feladatok az alábbi pályára vonatkoznak.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Melyik mezőn áll meg a csónak a megadott kódok hatására?   |  |  |  | | --- | --- | --- | | **Kérdés** | **Kód** | **Mező** | | A. | BB |  | | B. | JJB |  | | C. | BJBBJ |  |   Van egy teli bójánk (⚫). Melyik üres mezőre kell tenni ahhoz, hogy a **BJJ** kód hatására a csónak a megadott célmezőn álljon meg? Ha több megoldás is van, add meg az összeset, vesszővel elválasztva! (A nyilat és a hullámokat tartalmazó mezőkre nem tehető bója.)   |  |  |  | | --- | --- | --- | | **Kérdés** | **Cél** | **Teli bója helye** | | D. | C7 |  | | E. | H9 |  | | F. | H1 |  | |

2. feladat: Kijelző (25 pont)

Egy 7 szegmenses kijelzőn a betűket a következőképpen jelenítjük meg:



Rajzoló eszközünk (teknőc, macska, méhecske vagy amit szeretnétek) kezdetben a 7 szegmes bal alsó sarkában áll és felfelé (azaz északra) néz. Az E (előre), H (hátra), B (balra), J (jobbra) parancsokkal vezérelhetjük, a mozgások egy szegmensnyit lépnek, a fordulások 90 fokot fordulnak. Az L betűt pédául az J E H B E E parancs sorozattal rajzoltathatjuk ki.

Az alábbi 5 parancs sorozat betűket rajzol, add meg, hogy melyik parancs sorozat melyik betűt!

A. E E J E J E E H J E

B. J E H B E J E H B E J E

C. E E J E J E J E

D. J E B E B E J E J E

E. E J E B E B E B E

3. feladat: Függvény (18 pont)

A valami függvény egy szót kap paraméternek.

def valami(x):  
 if len(x)<2:  
 return 0  
 else:  
 if x[0]==x[1:][0]:   
 return 1+valami(x[1:][1:])  
 else:  
 return valami(x[1:])

Mit adnak eredményül a következő függvényhívások?

A. valami ”anna

B. valami ”panna

C. valami ”mississippi

D. valami ”aranka

E. valami ”annabella

F. Milyen paraméterre lesz 0 a függvény értéke?

G. Fogalmazd meg általánosan a függvény feladatát!

4. feladat: Szimbólumok (33 pont)

Az alábbi eljárások különböző szimbólumokat rajzolnak ki. Azonban ezekből hiányzik egy-egy utasítás. Ezek helyét egy üres dobozzal jelöltük. Add meg a hiányzó utasításokat úgy, hogy a mellékelt szimbólumokat rajzolja ki az eljárás! A teknőc kiinduláskor felfele néz, a tolla le van engedve.

|  |  |
| --- | --- |
| A:  def szimb1( h):  for i in range( 2):  turtle.forward( h/4)    turtle.forward( h/8)  turtle.right( 90)    for i in range( 180):  turtle.forward( 1)  turtle.right( 1)  for i in range( 2):  turtle.forward( h/4)    turtle.forward( h/8)  turtle.right( 90) | B:  def szimb2( h):  turtle.pendown()  turtle.forward( h/3)    turtle.forward( h/3)    turtle.forward( h/3)  turtle.penup()  turtle.backward( h)  turtle.left( 90)  turtle.forward( h/2)  turtle.right( 100)  turtle.pendown()  turtle.forward(h)  turtle.penup()  turtle.backward(h)  turtle.left( 100)  turtle.backward( h)    turtle.pendown() |
| C:  def szimb3( h):  turtle.forward( h)    turtle.forward( h\*math.sqrt(2))  turtle.left( 135)  turtle.forward( h)  for i in :  turtle.backward(h/6)  turtle.left( 90 )  turtle.forward( h/20)    turtle.forward( h/20)  turtle.right( 90) | |