

TEKNŐCGRAFIKA A SNAP! PROGRAMOZÁSI KÖRNYEZETBEN

1. BEVEZETÉS

1.1. A SNAP!

A Snap! egy ingyenes blokkalapú oktatási célú programozási környezet, amelyben animációk és játékok programozásán kívül technőcgrafikai programok is készíthetők. Online és offline is változata is van. A honlapja alapvetően angol nyelvű, de a programozási felülete és a programozási nyelve magyarul is használható.

A Snap! honlapja: <https://snap.berkeley.edu/>

1.2. A SNAP! ÁLTALÁNOS ELŐNYEI

Tekintsük át először a Snap! legfontosabb általános (nem csak a technőcgrafikához kapcsolódó) előnyeit az Imagine Logóhoz és a Scratch-hez képest:

	Imagine	Scratch	Snap!
– a népszerű asztali operációs rendszerek mindegyikén futtatható (a böngészőben fut);	x	✓	✓
– online és offline változata is van;	x	✓	✓
– az offline változatot nem kell telepíteni (az is a böngészőben fut);	x	x	✓
– a programkód blokkalapú, ezért csak szintaktikailag helyes programkódok állíthatók elő;	x	✓	✓
– a blokkalapú programkód a billentyűzettel is összeállítható, gyorsabban, mint az egérrel.	x	x	✓

1.3. A SNAP! ELŐNYEI TEKNŐCGRAFIKAI SZEMPONTBÓL

Az alábbiakban a Snap! technőcgrafikai többletlehetőségei olvashatók a Scratch-hez képest:

	Scratch	Snap!
– az alapértelmezett szereplő felülnézeti;	x	✓
– beállítható a „rajzlap” mérete;	x	✓
– zárt területek kitölthetők valamilyen színnel;	x	✓
– szöveg írható a „rajzlapra”;	x	✓
– készíthetők visszatérési értékkel rendelkező függvények;	x	✓
– eljárás (vagy függvény) paramétere lehet lista.	x	✓

2. A SNAP! PROGRAMOZÁSI FELÜLETE

2.1. A SNAP! HASZNÁLATA ONLINE ÉS OFFLINE

A Snap! online futtatásához elegendő a snap.berkeley.edu címen elérhető honlapjának a felső menüsorában található *Run Snap!* menüpontra kattintani. Az online változat használatához nem szükséges regisztrálni, ugyanakkor regisztrált felhasználóként a felhőben tárolhatók az elkészített programok. Regisztrálni a felső menüsor *Join*, bejelentkezni pedig a *Log In* menüpontjával lehet.

A Snap! offline változatának a letöltési linkje a honlapjának az alsó linklistájában található, de közvetlenül is elérhető a snap.berkeley.edu/offline címen. Nem szükséges telepíteni, az online változathoz hasonlóan a böngészőben fut, és attól egyetlen szempontból különbözik: az elkészült munkák nem menthetők el a felhőbe, de természetesen fájlként igen.

2.2. ÚJ PROJEKT ÉS A NYELV BEÁLLÍTÁSA

A Snap! elindításakor máris hozzáláthatunk egy új üres program elkészítéséhez. A Snap!-ben a programokat projekteknek nevezzük. Új projektet kezdeni, egy létező projektet megnyitni (a felhasználói fiókunkból vagy a számítógépről) vagy az aktuális projektet elmenteni (a felhasználói fiókunkba vagy a számítógépre) a fájl (📁) menüben lehet.

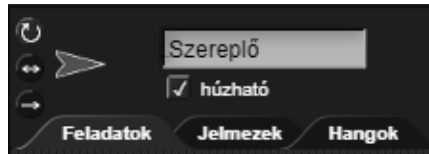
Ha a programozási felület angol nyelven jelent meg, akkor a felső menüsáv beállítások (⚙️) menüjében elérhető *Language* (🌐) menüpontra kattintva kiválaszthatjuk a magyar nyelvet (*Magyar*).

2.3. A „RAJZLAP”

A programozási felület jobb felső (a képen fehér háttérű) területén találhatjuk a játékeret, amelyet a teknőcgrafika során rajzlapként fogunk tudni használni. Alapértelmezetten 480×360 képpont méretű, ezt azonban megváltoztathatjuk a beállítások (⚙️) menüben található *Játéktér mérete...* menüponttal. Az origó a játéktér közepén van, és a játéktér szélei nem érnek össze (ha egy szereplő elhagyja a játékeret például a jobb oldalán, akkor nem fog megérkezni a bal oldalra).

2.4. A RAJZOLÁST VÉGZŐ SZEREPLŐ(K)

A játéktéren belül egy nyílalakú alapértelmezett szereplőt találhatunk az origóban, jobbra néző irányban. A nyíl hegyénél található az alapértelmezetten felemelt, szürke színű és 1 pixel vastagságúra beállított tolla. A nyíl színe megegyezik a toll mindenkor színével. A szereplőt átnevezhetjük a programozási felület középső oszlopjának a felső részén (1. ábra). Itt a *húzható* jelölőnégyzet kikapcsolásával elérhetjük, hogy a szereplőt az egérrel ne lehessen a játéktéren vonszolni.



1. ábra A szereplő nevének a helye

Szükséges esetben további szereplőket hozhatunk létre a játéktér bal alsó sarkánál található három gomb közül a bal szélsőre kattintva (2. ábra).

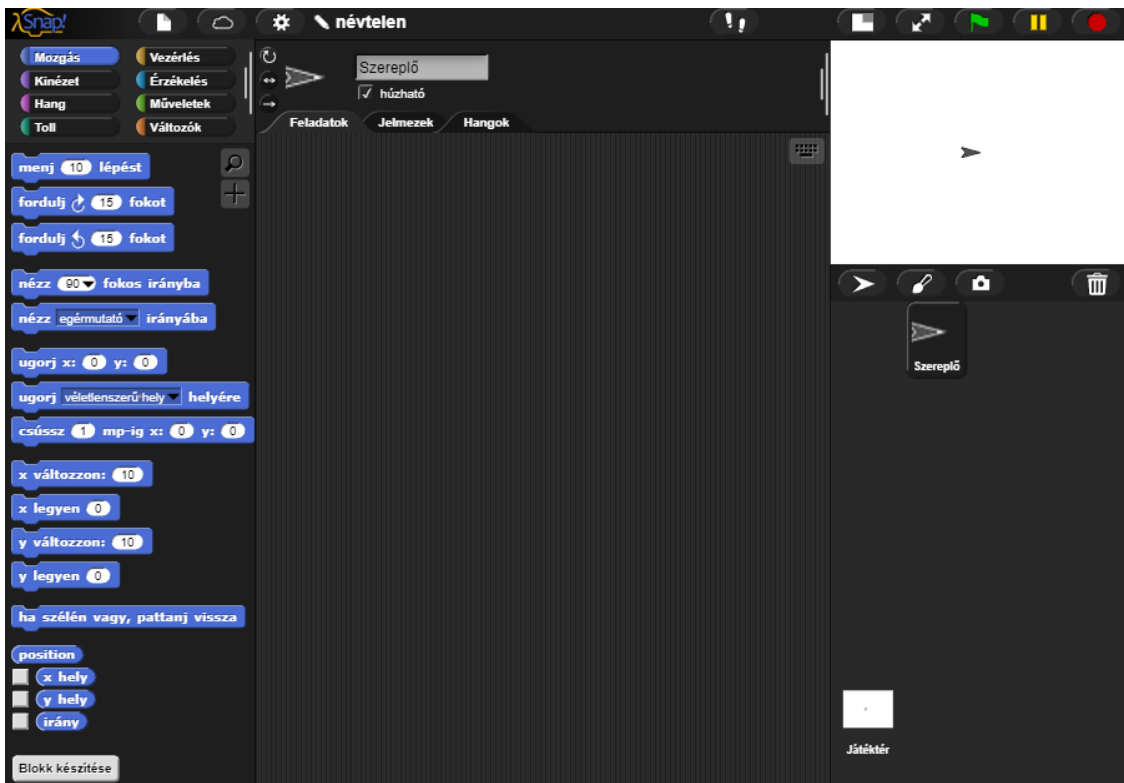


2. ábra További szereplők létrehozása

Több szereplő esetén a játéktér alatti szereplőlistában tudjuk kiválasztani azt, amelyiket aktuálisan szerkeszteni vagy programozni szeretnénk.

2.5. A PROGRAMOZÁSI TERÜLET ÉS A BLOKK-KÉSZLET

A programozási felület középső oszlopában a *Feladatok* fülre kattintva hozhatjuk létre az aktuálisan kiválasztott szereplő eljárásait. (Továbbá a *Jelmezek* fülön szerkeszthetjük a lehetséges kinézeteket, a *Hangok* fül mögött pedig az általa megszólaltatható hanghatásokat, ezekre azonban a teknőcgrafikához nem lesz szükségünk.) Ha egy szereplő esetén a *Feladatok* fül van kiválasztva, akkor a fül alatti területen a szereplőhöz tartozó programozási terület látható (3. ábra).



3. ábra A programozási felület részei balról jobbra: blokk-készlet, programozási terület, játéktér és szereplőlista

A felhasználói felület bal oldali oszlopában található a blokk-készlet, amelyben nyolc kategóriába sorolva érhetjük el a programozáshoz használható nyelvi elemeket, amelyek között egyebek mellett a szereplő állapotát (például a helyét és irányát) megváltoztató utasítások, a szereplő állapotát lekérdező blokkok, vezérlési szerkezetek és matematikai műveletek is megtalálhatók. A blokkok különlegessége, hogy a formájuknak köszönhetően csak szintaktikailag helyesen illeszthetők egymáshoz.

2.6. BLOKKOK ELHELYEZÉSE AZ EGÉRREL ÉS A BILLENTYŰZETTEL

A blokkok az egérrel áthúzhatók a programozási területre, illetve egy blokk vagy egy blokksozrat törölhető a programozási területről a blokk-készletbe történő visszahúzásukkal.

Ezenkívül a programozási terület jobb felső sarkában található billentyűzet (☞) ikonra történő kattintást követően a billentyűzettel is elhelyezhetők a programozási területen belül blokkok. A begépeltek kifejezésnek megfelelő blokkok a blokk-készlet helyén jelennek meg, amelyek közül a *fel* és a *le* nyíl gombokkal választhatunk, a kiválasztott blokkot pedig az *enter* billentyűvel szűrhetjük be a kódnak a vastag fehér vonallal, mint kurzorral jelölt részére (4. ábra).



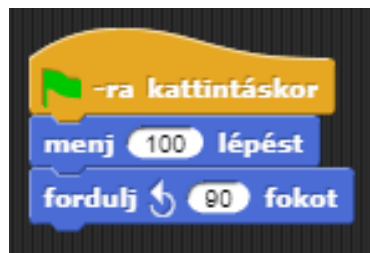
4. ábra A kód létrehozása gépeléssel

A kurzort a *fel* és a *le* nyíl gombokkal mozgathatjuk a blokkok között, egy blokkon belüli paramétereken pedig a *bal* és a *jobb* nyíl gombokkal haladhatunk végig. A kurzorral kijelölt paramétert az *enter* billentyű lenyomásával tehetjük szerkeszthetővé, és az *enter* újabb lenyomásával megszüntethetjük a szerkesztési állapotát, vagy a *Tab* billentyűvel rögtön a következő paraméterre ugorhatunk (szerkesztési állapotban).

2.7. EGY UTASÍTÁS VAGY UTASÍTÁSSOROZAT VÉGREHAJTÁSA

Az aktuálisan kiválasztott szereplő két esetben hajt végre egy utasítássorozatot: vagy ha az utasítássorozatra kattintunk a programozási területen belül, vagy ha egy eseményhez van

hozzárendelve egy eseménykezelő blokk segítségével (5. ábra). Egyetlen utasítás végrehajtásához közvetlenül a blokk-készletben is rákattinthatunk.



5. ábra Egy utasítássorozat eseményhez rendelése

Teknőcgrafikai programok esetén nem eseménykezelő, hanem névvel és paraméterekkel ellátott eljárások létrehozása a jellemző.

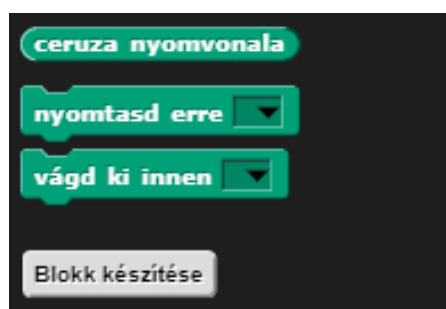
3. TEKNŐCGRAFIKA PROGRAMOZÁSA A SNAP!-BEN

3.1. A TEKNŐC KEZDŐÁLLAPOTBA HOZÁSA ÚJ ELJÁRÁS LÉTREHOZÁSÁVAL

A teknőcgrafikai eljárások létrehozása előtt célszerű egy olyan **kezdőállapot** x y nevű eljárást bevezetni, amely egyrészt letörli a játékterről a korábbi rajzokat, másrészt pedig a szereplőnek a rajzolás szempontjából lényeges tulajdonságait az általunk megadott kezdőértékekre állítja be. A szereplő kezdőhelyét az eljárás x és y paraméterével szeretnénk majd szabályozni.

3.1.1. AZ ELJÁRÁSHOZ TARTOZÓ BLOKK LÉTREHOZÁSA

Az eljáráshoz tartozó blokk létrehozásához kattintsunk a *Toll* blokk-kategóriára, majd az oszlopban megjelenő, a toll használatával kapcsolatos halványzöld blokkok listáját követő *Blokk készítése* szürke gombra (6. ábra). A kiválasztott blokk-kategória csak az új blokkunk színét fogja meghatározni, kifejezhetjük vele, hogy elsősorban melyik kategóriába illik a leendő blokkunk tevékenysége, de a blokkunk kategóriáját később módosíthatjuk.



6. ábra Blokk készítése gomb

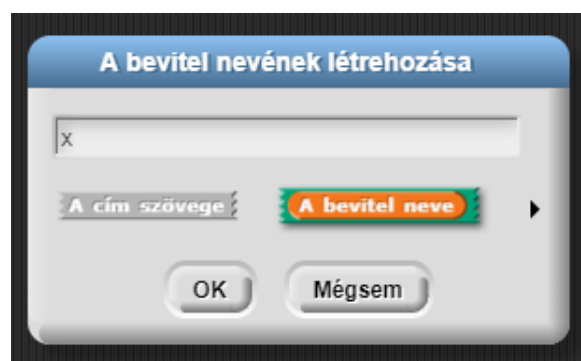
A megjelenő ablakban adjuk meg a *kezdőállapot* blokknevet, majd kattintsunk az *OK* gombra. Ezáltal létrejön a **kezdőállapot** nevű halványzöld blokk a *Toll* blokk-kategórián belül, valamint automatikusan megnyílik a blokk szerkesztő ablak (7. ábra).



7. ábra A blokszerkesztő ablak

3.1.2. A BLOKK PARAMÉTEREINEK A LÉTREHOZÁSA

A blokk szövege jelenleg egyetlen úgynevezett címkét, a *kezdőállapot* szót tartalmazza. Címke elé vagy mögé újabb címke vagy paraméter adható a megfelelő helyen található *+* jelre kattintással. Kattintsunk a *kezdőállapot* szót követő *+* jelre, és adjuk meg a megjelenő párbeszédablakban az *x* paraméternevet (8. ábra). Ebben az ablakban alapértelmezetten *A bevétel neve* opció az aktív, amely esetén paraméter adható hozzá a blokk szövegéhez. Címke hozzáadásához *A cím szövege* lehetőséget kellene kiválasztanunk. Címkék és paraméterek tetszőleges sorrendben követhetik egymást, tehát például létrehozható lenne egy **kör** **r** **sugárral** szövegű blokk is, amelyben az **r** lenne a paraméter, előtte és utána pedig egy-egy címke állna.



8. ábra Új paraméter nevének a megadása

Ezután ismét kattintsunk az eljárás fejsorának az utolsó *+* jelére, és adjuk meg ezúttal az *y* paraméternevet (9. ábra).



9. ábra Az eljárás blokkja egy címkével és két paraméterrel

3.1.3. A BLOKK KÓDJÁNAK A LÉTREHOZÁSA

Az eljáráshoz tartozó blokk kódja a blokkszerkesztő ablakon belül a sárga fejsor alatt hozható létre a megfelelő blokkok egymásután illesztésével. Az ablak mérete a jobb alsó sarkánál fogva megnövelhető.




Blokkok továbbra is behúzhatók a blokk-készletből, vagy a blokkszerkesztő ablak jobb felső sarkában található billentyűzetikorra (☰) kattintást követően a billentyűzettel is dolgozhatunk.

Hozzuk létre a blokk kódját az alábbi lépéseket követve.

1. Helyezzük először az **ugorj x: ... y: ...** blokkot az eljárásba, amely a szereplőt a benne paraméterként megadott helyű pontra fogja áthelyezni. Mivel azt szeretnénk, hogy a **kezdőállapot** blokk **x** és **y** paraméterével a szereplő kezdőhelyét lehessen meghatározni, az **ugorj** blokk két paramétereként a **kezdőállapot** blokk két paraméterét kell megadnunk. Ezt például egérrel a következőképpen valósíthatjuk meg: fogd és vidd módszerrel húzzuk a blokk fejsorából az **x** majd az **y** paraméternevet az **ugorj** blokk megfelelő paraméterhelyére (10. ábra). A paraméterek is begépelhetők a billentyűzettel.



10. ábra Az eljárás paraméterének az elhelyezése egérhúzással az eljárás kódjában

- Adjuk az eljáráshoz a **nézz**  **fokos irányba** blokkot, amelyben a szereplő abszolút iránya begépeléssel vagy a lenyíló listára kattintással is megadható. A 0 fokos irány a képernyő szerinti felfelé iránynak felel meg.
- Csatlakoztassuk ezután az eljáráshoz a **töröld a rajzokat** utasítást, amely eltünteti a játéktérről a tollakkal korábban készített rajzokat.
- Most egészítsük ki az eljárást a **tollméret legyen**  utasítással. A tollméretet érdemes a Snap!-ben mindig legalább 2 képpont vastagságúra állítani, ugyanis területek színes kitöltése esetén (a játéktéren a pixeleződés elkerülése miatt automatikusan végrehajtott elmosás miatt) a területeket határoló vonalak kismértékben elvékonyodhatnak.
- Helyezzük ezután a **tollszín legyen**  blokkot az előző után. Ebben a blokkban a véletlenszerűen beállított színmintára kattintva egy megjelenő palettáról választhatunk egy feketehez közeli (vagy bármilyen más) árnyalatot. (A *toll színe* című részben olvashatók részletesebben a tollszín megadásának a módjai.)
- Végül a **tollat le** utasítás használatával érhetjük el, hogy a **kezdőállapot** blokk használatát követően a szereplő tolla mindenképpen lent legyen (11. ábra).

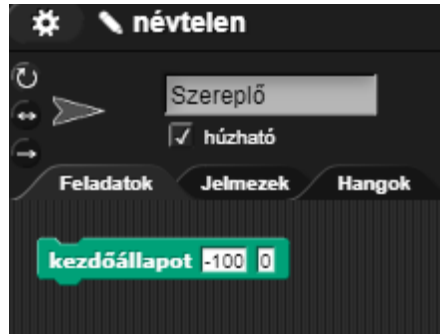


11. ábra A kész „kezdőállapot” eljárás

Miután elkészítettük a szükséges programkódot, a blokkszerkesztő ablakot az *OK* gombjára kattintva bezárhatjuk. Ehhez az ablakhoz később visszatérhetünk például úgy, hogy a **kezdőállapot** blokkra kattintunk jobb egérgombbal akár a blokk-készletben, akár a programon belül, és a helyi menüből a *szerkesztés...* menüpontot választjuk.

3.1.4. A BLOKK HASZNÁLATA

A blokkot kipróbálhatjuk például úgy, hogy egy példányát elhelyezzük (a megfelelő szereplő kiválasztását követően) a programozási területen, megadjuk a két paraméterének az értékét, és a blokkra kattintunk (12. ábra).



12. ábra Egy (eseményhez nem kötött) eljárás futtatásához elegendő kitölteni a paramétereit majd rákattintani

3.2. A SZEREPLŐ MOZGATÁSA

A szereplőt az aktuális irányába a paraméterként megadott távolsággal előre mozdítani a **menj** **lépést** utasítással lehet, a megadott szögben elforgatni pedig a **fordulj** **fokot** utasítással (a paraméterhely előtti nyíl irányba fejezi ki a fordulás irányát). A hátrafelé mozgatás a **menj** **lépést** utasítás negatív paraméterezésével lehetséges, vagy létrehozható (például) egy **tolass** **lépést** nevű eljárás is.

Különleges esetekben hozzá kell férnünk a szereplő aktuális helyét és irányát meghatározó számokhoz az **x hely**, az **y hely** és az **irány** kifejezések felhasználásával. Szintén speciális feladatok esetén szükség lehet (a kezdőállapotba hozáson kívül is) a szereplő helyének és irányának abszolút beállítására az **ugorj x:** **y:** illetve a **nézz** **fokos irányba** utasításokkal.

Érdeemes tudni, hogy a szereplők a nagyszámú mozgatóutasításokat alapértelmezetten viszonylag lassan hajtják végre (ezzel könnyítve a kezdő programozók számára az animációkészítést). Tipikusan a körívek rajzolása tart sokáig, amely felváltva nagyon sok előre lépést és elfordulást igényel. Az ilyen helyzetekben a kritikus programrészt a **Vezérlés** blokk-kategóriában megtalálható **gyorsítva** blokkba helyezhetjük, ezáltal nagymértékben felgyorsítva annak végrehajtását (13. ábra).



13. ábra A „gyorsítva” blokkba helyezett programrész futása közben a Snap! nem rajzolja ki a szereplőt, ezáltal lényegesen rövidül a programrész végrehajtási ideje

3.3. A TOLL ÁLLAPOTA

A szereplő a tollát képes letenni és felvenni a **tollat le** és a **tollat fel** utasítással. A toll ilyen értelemben vett állapota lekérdezhető a logikai értékű **toll lent?** kifejezéssel.


3.4. A TOLL VASTAGSÁGA


A toll vastagsága beállítható a **tollméret legyen** utasítással. A tollméretet érdemes a Snap!-ben mindig legalább 2 képpont vastagságúra állítani, ugyanis területek színes kitöltése esetén (a játéktéren a pixeleződés elkerülése miatt automatikusan végrehajtott elmosás miatt) a területeket határoló vonalak kismértékben elvékonyodhatnak. A **tollméret változon** utasítás a toll méretét a megadott számmal növeli, vagy negatív szám esetén csökkenti. A toll vastagsága lekérdezhető a **toll** kifejezéssel, amelynek a lenyíló listájából a *méret* szót kell kiválasztani.




3.5. A TOLL SZÍNE

A toll színét a *Toll* blokk-készlet alapértelmezett blokkjaival kétféleképpen állíthatjuk be. A **tollszín legyen** utasítással a paraméterére kattintva egy színátvívóval kiválasztott színre állíthatjuk be a tollszínt. Hátránya, hogy a színminta csak konstans lehet, tehát ez az utasítás nem paraméterezhető például színkóddal vagy színnévvel.

A másik lehetőség a HSL színmodell alkalmazása. A **toll árnyalat legyen** utasítással a toll színárnyalatát állíthatjuk be 0-tól 100-ig terjedő értékkel (a 0 a piros, a 33 a zöld, a 66 a kék és a 100 ismét a piros), a **toll színtelítettség legyen** utasítással az árnyalat telítettségét szintén egy 0-tól 100-ig terjedő értékkel (a 0 a szürkének a 100 a teljesen telített

árnyalatnak felel meg), a **toll világosság legyen**  utasítással pedig az árnyalat világossága szabályozható ismét egy 0 és 100 közötti értékkel (a 0 a fekete). Ezen utasítások előnye, hogy paraméterezhetők számmal. Mindhárom említett utasításnak elérhető egy olyan változata, amely az értéket nem beállítja, hanem növeli (vagy negatív előjelű szám esetén csökkenti) a paraméterként megadott számmal. Ezen változatokban a *legyen* szó helyett a *változon* szó szerepel.



A toll árnyalata, színtelítettsége és világossága lekérdezhető a **toll**  kifejezéssel, amelynek a lenyíló listájából a megfelelő szót kell kiválasztani.

A toll színét beépített színnevekkel és színkódokkal is beállíthatjuk az Imagine Logóban megszokottakhoz hasonlóan, ehhez azonban a fájl  menü *Modulkönyvtárak...* menüpontjában importálni kell a *Crayons* nevű modult. Ezt követően a *Toll* blokk-kategóriában elérhetővé válik a **set pen to crayon**  utasítás, amelynek a paraméterére kattintva angol színnevek majd azokon belül színárnyalatnevek (és eléjük írt színkódok) közül választhatunk. Az utasítás színneveket és színkódokat is elfogad paraméterként. A **change crayon by**  utasítással növelhető (vagy negatív előjel esetén csökkenthető) a színkód a paraméterként megadott számmal. A **pen crayon** kifejezéssel lekérdezhető a szereplő tollának az aktuális színkódja.

3.6. TERÜLET KITÖLTÉSE

A szereplővel az alatta lévő egyszínű terület kitölthető a **kitöltés** utasítással. A szereplőnek nincs külön kitöltőeszköze, a toll aktuális színével végzi a kitöltést.




3.7. SZÖVEG KIÍRÁSA

Szöveget a szereplő az aktuális helyére az aktuális irányába az aktuális tollszínével az **írd**  **mérettel**  tud kiírni. Az első paraméterben adható meg a kiírandó szöveg, a másodikban a betűméret.

3.8. VEZÉRLÉSI SZERKEZETEK ÉS MATEMATIKAI-LOGIKAI MŰVELETEK

A teknőcgrafikához szükséges számlálós ciklus és elágazás elérhető a *Vezérlés* blokk-kategóriában, a megszokott matematikai és logikai műveletek pedig a *Műveletek* kategóriában. Az összeadás, illetve a szorzás műveletének a blokkjában a tagok, illetve a tényezők száma növelhető vagy csökkenthető a blokk jobb oldalán található nyilakra kattintva.

3.9. LISTA BEJÁRÁSA

A Snap! listákat is kezel, az ezzel kapcsolatos blokkok a *Változók* kategórián belül találhatóak. Listát az elemek felsorolásával megadni a **lista**  kifejezéssel lehet. A blokk jobb oldalán található nyilakra kattintva növelhető vagy csökkenthető az elemszám. Eljárás paramétere lista is lehet. Egy lista elemenkénti feldolgozásához elegendő a **minden elemre**  **ebből** 

blokkot használni, amellyel elemenként bejárható a lista. Az alábbi példában az elem feldolgozását külön segédeljárás végzi (14. ábra).



14. ábra Lista feldolgozása elemenként