

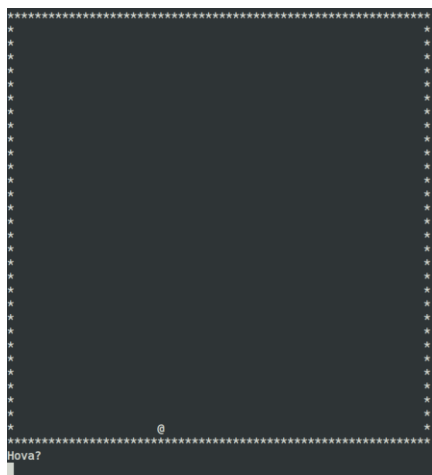
1. Snake

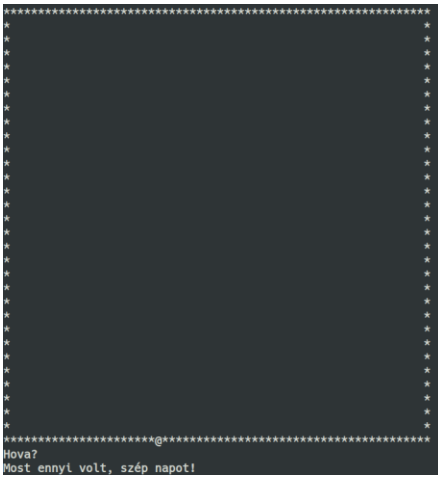
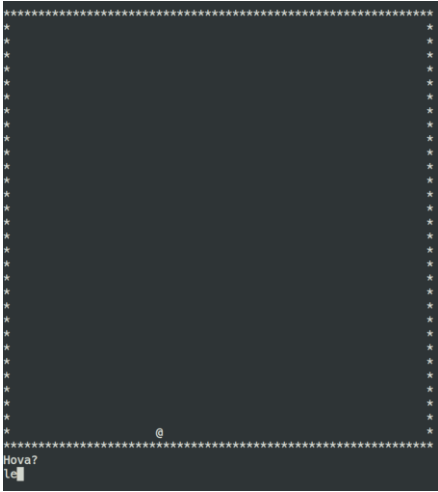
A kígyós videójátékok a kezdetektől nagy népszerűségnek örvendenek, az első változatot még 1976-ban hozták létre. A feladatod, hogy egy hasonló játékot hozz létre, amiben egy kis pitont kell irányítanod. Hogy a pitonod életszerűen viselkedjen, a legideálisabb, ha a Python programozási nyelvet használod :-).

Az alábbi szabályokat vedd figyelembe a játék elkészítésekor:

- a játék egy 60 x 30-as pályán játszódik, ez az a tér, ahol a piton mozoghat,
- a pitonod egy “@” karakterrel keltsd életre, amit a játék indulásakor egy véletlenszerű ponton helyezz el a pályán,
- a pálya szélét (a játéktéren kívül) kerítés védi, amit a “*” karakterrel jeleníts meg,
- amikor a játék elindul, rajzold ki a pályát minden elemmel együtt, és jelenítsd meg a “Hova?” feliratot, amivel a pitonunk kérdezi a játékostól, hogy mit csináljon,
- a pitonnal magyarul kell tudnunk beszélgetni, de csak az alábbi szavakat kell megtanulnia és megértenie:
 - balra
 - jobbra
 - fel
 - le
 - meguntam
- a “meguntam” szóra jelezzük, hogy elégünk lett a játékból és ideje pihenni menni (befejezhetjük a program futását),
- a többi szóra értelemszerűen a megfelelő mozgást kell végrehajtani a pitonunkkal,
- egyéb utasítást nem kell értelmezni,
- ha a piton hozzáér a kerítéshez, hagyjuk vele abba a játékot (befejezhetjük a program futását),
- mielőtt a játékos kilép, illedelmesen köszönjünk el a kis pitontól a “Most ennyi volt, szép napot!” felirattal.

Példa:





2. Dobble generátor

A Dobble nevű társasjáték egy kártyapakliból áll, melyben minden kártyán n darab különböző ábra található. Bármely két kártyát választva a pakliból igaz az állítás, hogy a két kártyán egy és csakis egy azonos ábra látható. Egy olyan pakli esetén például, ahol $n=3$, két kártya lehet mondjuk a következő: (ágy, asztal, TV) - (fagyi, zokni, asztal). Itt az asztal a közös, a többi mind különböző.

A feladat egy olyan program készítése, mely egy paraméterül kapott n számhoz tartozó szabályos Dobble paklit generál. A grafikus ábrákat most természetes számok fogják helyettesíteni. A program kimenete soronként egy kártya tartalma kell legyen, a kártya "ábráit", azaz a számokat szóközzel kell elválasztani.

Például $n=3$ paraméterrel indítva, egy lehetséges szabályos kimenet a következő:

- `> ./dobble.py 3`
- 1 2 3
- 1 4 5
- 1 6 42
- 2 4 6
- 2 5 42
- 3 4 42
- 3 5 6

Fontos még megjegyezni, hogy nem lehet minden kártya-pár közös eleme ugyanaz az érték, így a következő megoldás érvénytelen (itt a közös érték mindig az 1):

- 1 2 3
- 1 4 5
- 1 6 7
- 1 8 9
- 1 10 11
- 1 12 13
- ...

A cél a minél nagyobb kártyaszámú pakli generálása.

3. Pizzarendelő chatbot

A pizzarendelés sokunk napi tevékenysége. Mivel nem mindig ugyanazt fogyasztjuk, ezért a kiválasztás néha nehézkes: a pizzabolt listáját elő kell keresni, végig kell gondolnunk, hogy épp mit szeretnénk, esetleg plusz feltéteket kérünk, akár innivalót is rendelnénk. Ez akár 10 percig is tarthat, miközben gyomrunk már szaporán korog.

A pizzarendelést azonban el lehet készíteni chatbot segítségével is, ahol elég annyival kezdeni, hogy sonkás, kukoricás hagymás pizzát szeretnék, majd a bot 2-3 további kérdés segítségével finomítja a pontos rendelést.

A feladat egy pizzarendelő chatbot implementációja Pythonban, ami írott emberi nyelven kommunikál a rendelővel egy egyszerű felületen és a rendelés végén előáll egy rendelési állomány, amiben benne vannak a megrendelt tételek és a végső ár. A választék másodlagos (2-3 pizza, 2-3 plusz feltét, 2-3 innivaló elegendő).

Példa:

that all thx

Chat!

User:

Assistant: Hello! Welcome to our pizza restaurant. What can I get for you today?

User: wanna pizza with eggplant

Assistant: Great choice! Would you like a small, medium, or large eggplant pizza?

User: large, and a coke as well

Assistant: Sure thing! So that's a large eggplant pizza and a coke. Is that all for your order or to add anything else?