

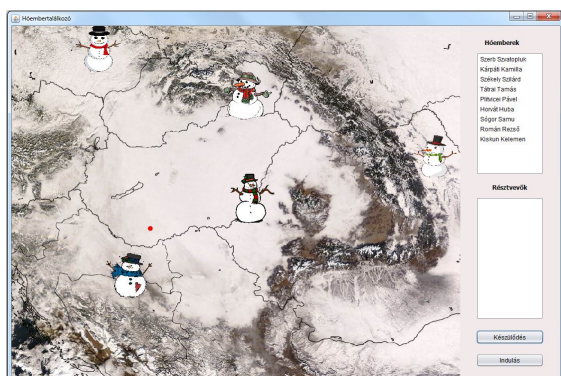
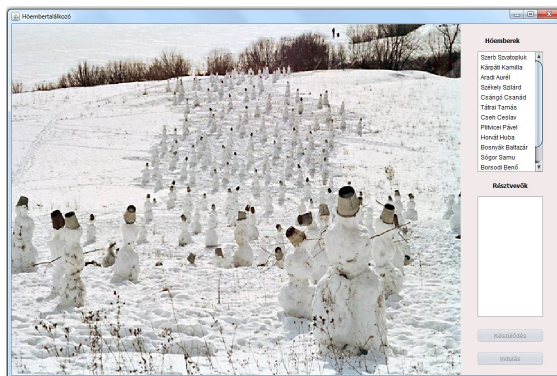


## Hóember-találkozó

Állítólag szilveszterkor nemzetközi hóember-találkozó volt a Mecseken. Ezt ugyan az újság megpróbálta cáfolni, de nincs igaza, épp most fogja szimulálni egy Java programmal. ☺

Induláskor a mellékelt felület látható, a jobboldalon fel vannak sorolva a találkozót fontolgató hóemberek. Adataikat egy adatfájl tartalmazza, soronként egy-egy név és két koordináta. A nevekhez a megfelelő sorszámú kép tartozik. A koordináták a „lakhelyének” koordinátái, innen indulnak majd. A fájlban adott pont a hóember talpának közepe.

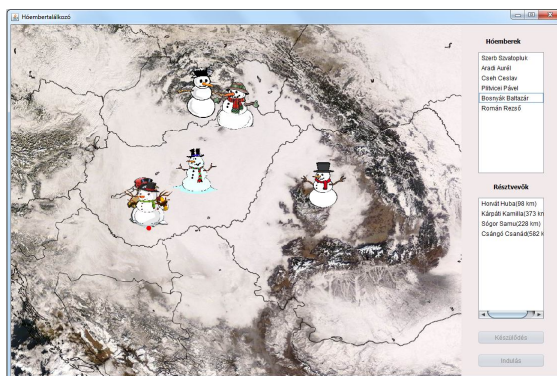
Induláskor mindkét gomb inaktív.



A rajzpanelre kattintva megváltozik a háttér, ekkor a készülődés gomb aktívvá válik, az indulás még inaktív.

A gombot megnyomva a felső listából kiválasztott hóemberek ráállnak az adott koordinátájú pontokra (a „lakhelyükre”). Ekkor nevük kikerül a hóemberek listájából.

Az indulás gomb akkor válik aktívvá, ha egyrészt a térképen kijelöljük a találkozó helyét, vagyis rákattintunk egy pontra (ekkor a pont körül megjelenik egy piros pötty), másrészt van legalább egy útra készülő hóember. A készülődés gombot akár többször is megnyomhatjuk, azaz többször is készülődhetnek, amíg az indulás gomb hatására el nem indultak. Az indulás gomb hatására a hóemberek véletlen sebességgel elindulnak a találkozó színhelyére.



Amíg mindegyik oda nem ér, addig ismét inaktív mindkét gomb, ha mindenki megérkezett, akkor a készülődés ismét aktívvá válik, az indulás pedig akkor, ha van legalább egy készülődő hóember.

Ha a hóember megérkezett, akkor neve kerüljön be a résztvevők listájába, mellette pedig zárójelben legyen feltüntetve az általa megtett távolság (Pithagorasz tétel ☺).

Ha megérkezik, akkor továbbra is legyen látható, de már ne mozogjon. (Rakja át az odaérkezettek listájába.)



Végül, ha még egyet kattintunk a felületre, akkor a rajzpanel jobb felső sarkából növekedni kezd, majd teljesen ellepi a rajzpanelt a hóemberek egyik lehetséges szomorú sorsát ábrázoló kép.

Javasolt méretek és adatok: belső felület: 1100 x 700 pixel, ebből a rajzfelület 900 pixel széles. A hóemberkép mérete: 100 x 100 pixel, a piros pötty sugara 5 pixel.

A mozgáshoz szükséges késleltetési idő alsó-felső határa: 20 és 50 ms, a kép mozgáshoz szükséges késleltetési idő 50 ms. De persze, ezektől el is térhet.

### Néhány megoldási ötlet:

A korábbiakhoz képest szinte semmi új nincs benne, pusztán annyika, hogy a hóember példányok létrehozását és elindítását szétválasztjuk egymástól. Figyeljen rá, hogy létrehozásakor frissítse a felületet. Az indító gomb hatására az összes készülődő hóembert elindítjuk.

A hóember mozgása a megszokott egyenes vonalú egyenletes mozgás. A kezdő- és végpont közötti távolságot könnyen ki lehet számolni, és ha célba ért, akkor a `toString()` metódusban ezt is hozzáadjuk a hóember nevéhez.

Még egy dologra kell figyelni: nem csak az úton lévőket hanem a helyszínen ácsorgókat is látjuk. Ez úgy oldható meg, hogy több listát is használunk. Az egyikben vannak a beolvasott példányok. Készülődéskor innen átkerülnek az úton lévőkhöz, őket látjuk majd mozogni, és ha megérkeztek, akkor nem kitöröljük őket, hanem átrakjuk az ott lévőkhöz. Kirajzoláskor e két utóbbi lista elemeit rajzoltatjuk.

Az indító gomb aktivitása azért kényes, mert ha nem tesszük inaktívvá, akkor ki kell védenünk, hogy ne próbáljuk meg kétszer elindítani ugyanazt a hóember-szálat.

A felületre kattintáskor figyelni kell, hogy hanyadszorra kattintottunk.

A végén megjelenő képet vagy egy külön szál animálja, vagy kezelhetjük a vezérlő osztályt is szálként, ekkor ő mozgatja. Ez akár egy sima `for` ciklus is lehet, amelyben a kép méretét kell növelni. A növekedést adó lépésköz a szélesség, illetve a magasság  $n$ -ed része (ha  $n$  lépésből áll a ciklus). Mivel jobbról balra növekszik, ezért végig kell gondolni, hogy hol van a kép aktuális bal felső sarka.