

## ELEVATOR MANAGER V.1.0.0



Programot és a dokumentációt írta:

Korek György - BPVKMQ

Másodéves gazdaságinformatika hallgató

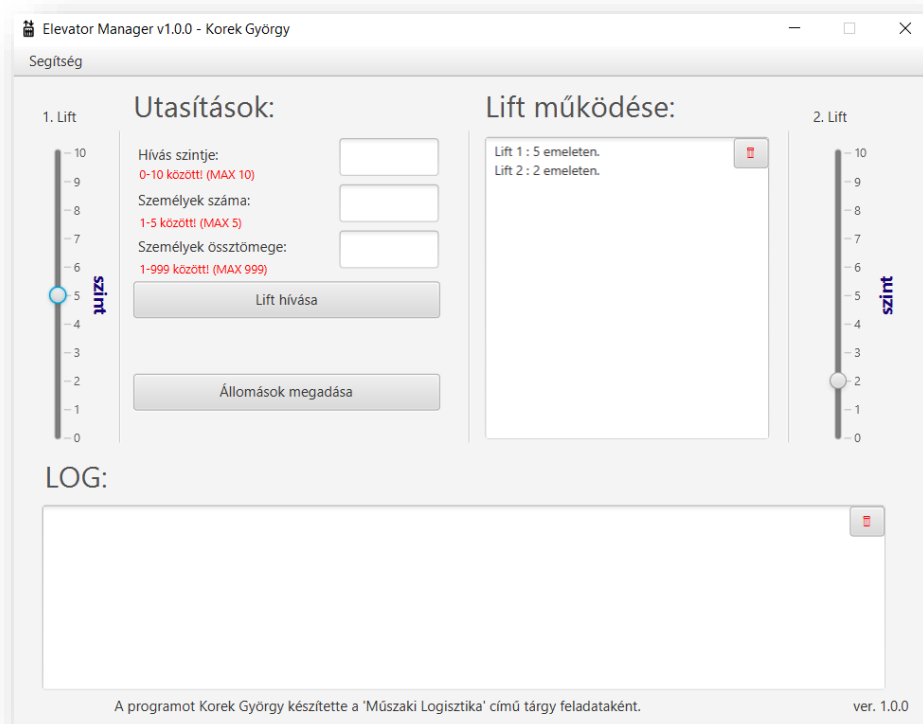
## PROGRAM CÉLJA

A program egy tíz emeletes épület 2 liftjét vezérli összehangoltan. A program mindig az ideálisabb helyzetben lévő liftet küldi a hívóhoz. Majd az utasokat a legoptimálisabb úton leszállítja.

## PROGRAM TECHNIKAI HÁTTERE

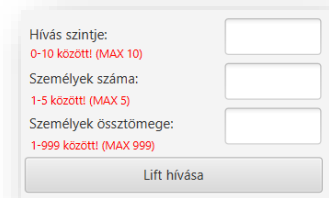
A program Java környezetben lett megírva, JDK 1.8.0\_231-es verziója alatt. A kezelőfelület (továbbiakban Gui) JavaFx alapú. A program az IntelliJ Community edition 2019.2.4, valamint az Oracle Scene Builder 2.0 verziójával készült. Jar fájlból, futtatható .exe fájlt a Launch4j-vel készítettem. Igyekeztem a széleskörű kompatibilitást szem előtt tartani, ezért esett a választásom az 8-as főverziójú JDK-ra, a JDK 14 & OpenFx JDK 14 és a Maven/Gradle kombináció helyett.

## PROGRAM KINÉZETE ÉS FUNKCIÓI

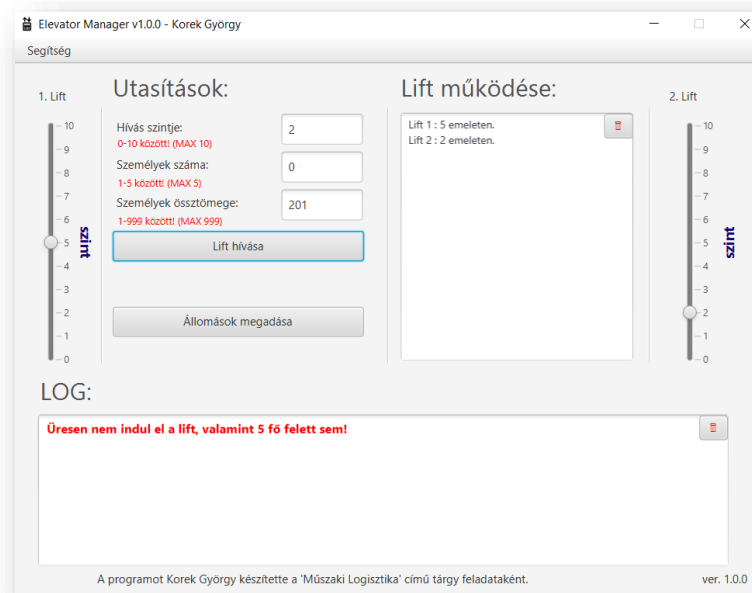


Elevator Manager az indítás után.

A kezelőfelületen 2 oldalt látható 1-1 „Slider” amin a liftek aktuális helyzete látható. Az liftek kezdő állapota véletlenszerűen határozódik meg, hogy életszerű helyzetet szimulálhasson már a kezdetektől. Kezdetben a felhasználó a „Hívás szintjét” a beszálló „Személyek számát” és a „Személyek össztömegét” adhatja meg. A hívás szintje azt az emeletet jelöli, ahonnan kezdeményezték a lift hívását. A beszálló személyek száma jelöli a liftbe beszálló személyek számát, akiket a liftnek el kell szállítania az általuk kezdeményezett céleleleltre. A személyek össztömege jelöli a beszálló személyek súlyát. Mivel életszerű helyzetet szimulál a program ezért beszálló személyeket 5 főben, míg a tömegüket 999 kilogrammban maximalizálja a program. E-fölött nem indul el a lift és erre figyelmezteti a felhasználót. Valamint a lift nem indul el üresen, azaz 0 beszálló személlyel.



Hívás szintje:   
0-10 között! (MAX 10)  
Személyek száma:   
1-5 között! (MAX 5)  
Személyek össztömege:   
1-999 között! (MAX 999)



Elevator Manager v1.0.0 - Korek György

Segítség

1. Lift

szint

10  
9  
8  
7  
6  
5  
4  
3  
2  
1  
0

**Utasítások:**

Hívás szintje:   
0-10 között! (MAX 10)  
Személyek száma:   
1-5 között! (MAX 5)  
Személyek össztömege:   
1-999 között! (MAX 999)

**Lift működése:**

Lift 1 : 5 emeleten.  
Lift 2 : 2 emeleten.

2. Lift

szint

10  
9  
8  
7  
6  
5  
4  
3  
2  
1  
0

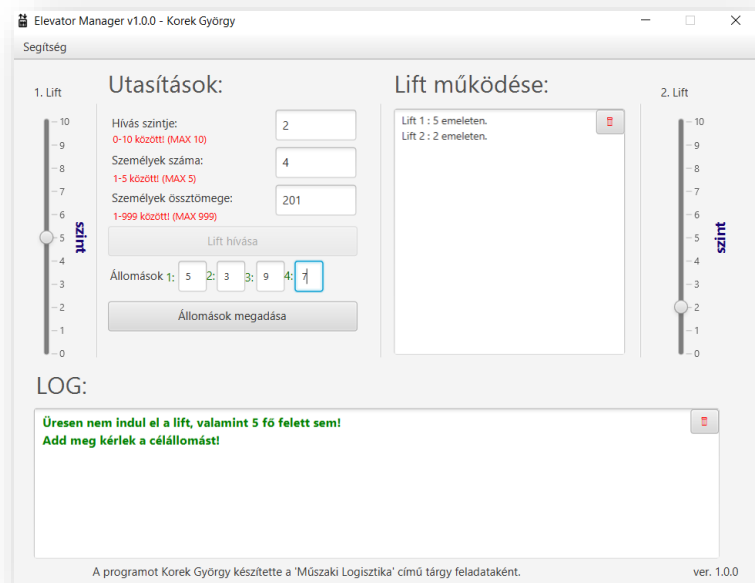
**LOG:**

Üresen nem indul el a lift, valamint 5 fő felett sem!

A programot Korek György készítette a 'Műszaki Logisztika' című tárgy feladataként. ver. 1.0.0

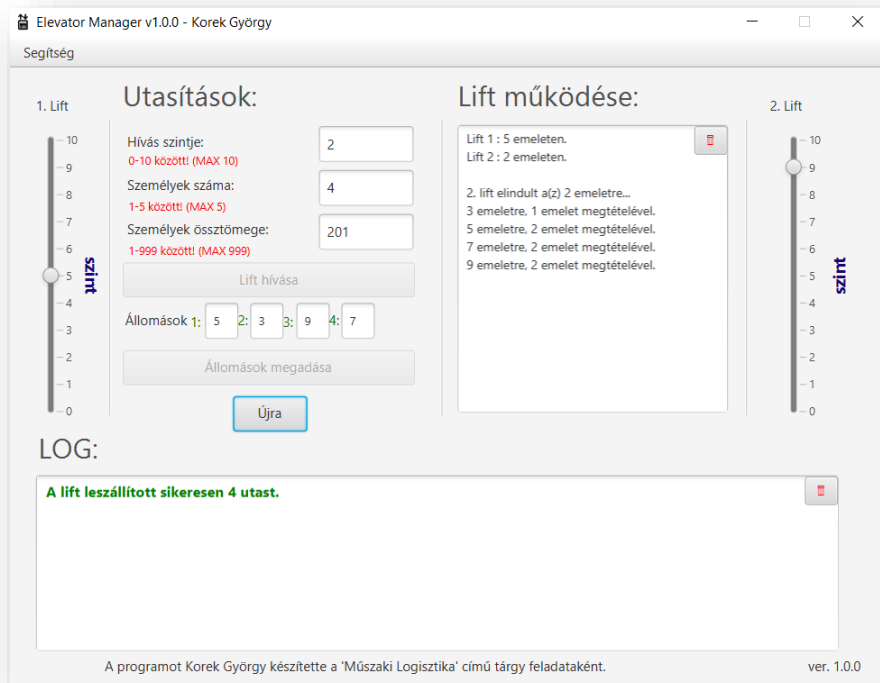
Helytelen inputok esetén a „LOG” felület alatt figyelmezteti a felhasználót, hogy mit nem adott meg helyesen.

Amennyiben helyes lifthívást indított a felhasználó, abban az esetben megnyílik egy új rész, ahol a felhasználó megadhatja az utasok célállomását.

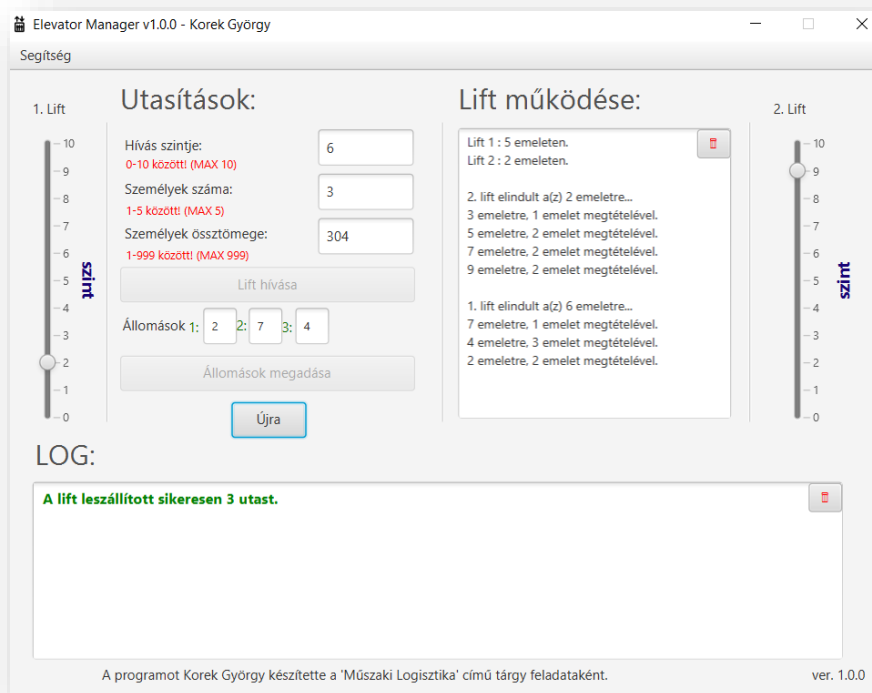


A személyek száma szerint nyílik lehetőség célállomás megadására. Így 4 személy esetén 4 állomást állíthat be a felhasználó, míg 1 ember esetében csak 1-et. Természetesen hibás input esetében itt is figyelmezteti a felhasználót, hogy hibás értéket adott meg.

Amennyiben a felhasználó 0-10 közötti értékeket ad meg, elindul a lifteknek a működése.



Mintán látható, hogy a hívás a második emeleten történt, így a közelebbi lift a 2-es lift volt. Ezt követően a program az előbb megadott célállomások szerint a legrövidebb úton leszállította az utasokat. Ennek folyamatát követhetjük a „Lift működése” felület alatt. Valamint a 2-es lift megváltozott emeletét követhetjük a fentebb ismertetett „Slidereken”. Amennyiben új hívást szeretnénk kezdeményezni, az „újra” gombra kell kattintania a felhasználónak.

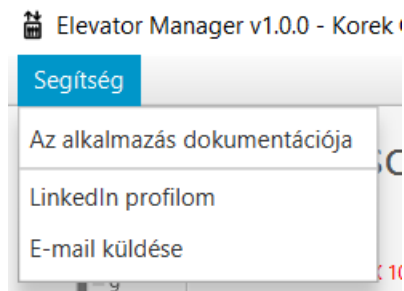


A mintaképen most 3 személlyel a 6. emeleten indított a felhasználó hívást. Ennek függvényében most az első lift ment értük és 3 célállomást lehetett megadni. És az 1-es lift helyzete is változott a Slideren.

A kuka ikonok segítségével ürítheti a felhasználó a szöveges log mezők tartalmát.



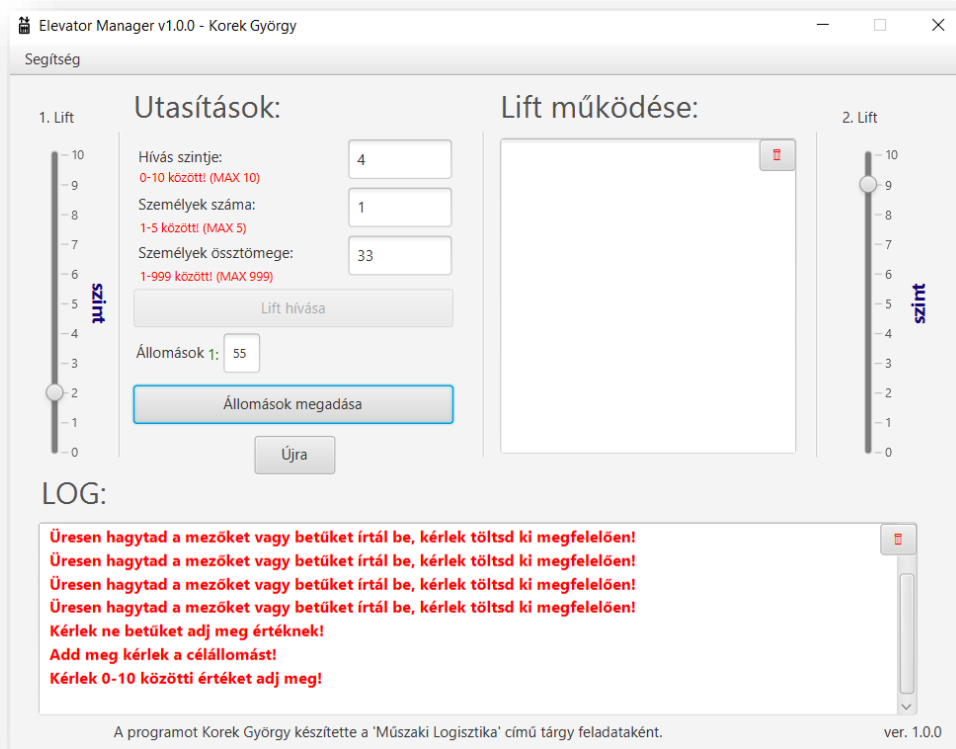
## SEGÍTSÉG MENÜPONT

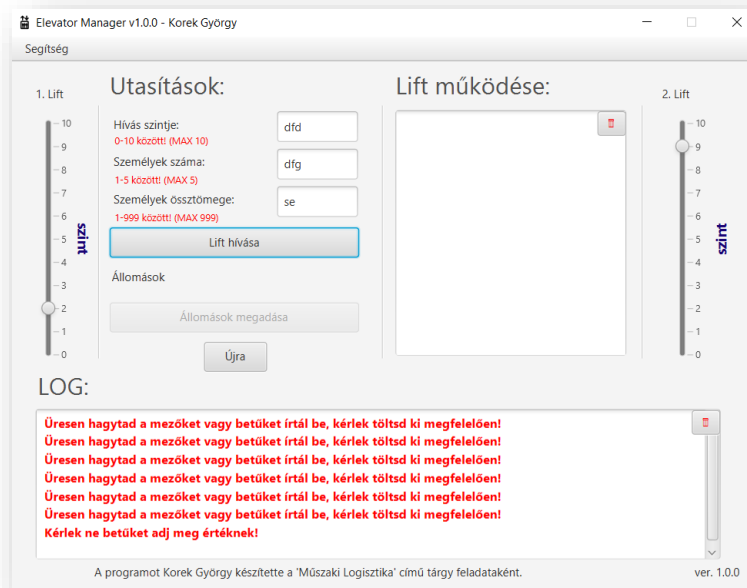


A segítség menüpontban 3 lehetőség adott. Az elsőre kattintva megnyílik a program dokumentációja. A második megnyitja a LinkedIn profilomat, míg a harmadik gomb a Gmail segítségével e-mail küldését kezdeményezi az előre megadott e-mail címre.

## HELYTELEN INPUTOK KEZELÉSE

A program valamennyi tesztelt helytelen értékmegadásra tesztelve van, amire figyelmezteti a felhasználót. Akár semmit, betűt, akár helytelen számot ad meg a felhasználó.





## FOLYAMATI MŰKÖDÉS

Hívás szintje = floorrequest (int), majd elist.get(x).getactualfloor();

Személyek száma = elist.get(x).getactualPerson(); (lift objektum egyik változója)

Személyek össz tömege = elist.get(x).getactualKG(); (lift objektum egyik változója)

Desztinációk = plist.get(x).getdestination(); (utas objektum egyik változója)

Kezdeti hívásra az a lift indul el, ami közelebb helyezkedik el. Azonos távolság esetén a lentebb elhelyezkedő lift indul. Program a fenti inputok után a hívás helyéhez képest meghatározza a legközelebb eső következő célállomást, majd elmegy oda és kirak 1 utast. Amint ez megtörtént a folyamat ismétlődik a legrövidebb úton, ameddig minden utast le nem szállított.

## A PROGRAM TÖRTÉNETE

A program először parancssoros (terminálos) Java programnak indult, hogy a megfelelő back-end kódot és logikát kidolgozhassam, és megfelelően tesztelhessem. A program objektum-orientáltan került kialakításra. A kizárólag back-end kód külön dokumentációval és forrásfájlokkal rendelkezik, mert annak kialakítása eltér a végeredményben létrehozott JavaFx alapú programtól. A fentebb ismertetett program, a kezelőfelület fxml alapú megtervezése után került a back-end kóddal való összhangba hozásra. A program a Budapesti Gazdasági Egyetem – Pénzügy és Számvitel karán, gazdaságinformatika szakon a „Műszaki Logisztika” elnevezésű tárgy beadandó feladataként került létrehozásra.

írta:

Korok György

2020.03.29.

## KIEGÉSZÍTÉS:

Csatoltam a beadandóhoz, a kezdetleges command-line alapú java forrását a programomnak. Úgy gondolom, ebből egyszerűbb és átláthatóbb a program működése és forrásának megértése. Arra kérem Tanár Urat, hogy a command-line forrást ennek tükrében tekintse meg, hiszen amikor késznek gondoltam összerakni a kezelőfelülettel, a fejlesztés további szakaszait és a hibajavításokat **már a JavaFx alapú programban végeztem el, így több exeption kezelést NEM tartalmaz, amit a végleges program már IGEN.** Így a beadandóm valódi értékét úgy gondolom az adja, ez csak segítség az értékeléshez és a program megértéséhez, amolyan flow check-nek alkalmas.