

## Menetrend

*TODO: Change name*

*Futaki Bálint, Vajda Simon*

*Felkészítő tanár: Kutas Tibor*

*Kecskeméti Bolyai János Gimnázium, Kecskemét*

### 1. Bevezetés

A tömegközlekedés sokak számára az elsődleges megoldás a mindennapi ingázásban, utazásokban. Azonban elég frusztráló, amikor nem tudjuk pontosan, hogy mikor és melyik járatra kell felszállnunk, hogy a leggyorsabban odaérjünk a kívánt úticélhoz. A forgalmasabb városokban az üzemeltetők honlapja vagy a Google Maps segítséget nyújt az útvonal tervezésében, de sajnos ez nem minden városban elérhető. Az mi lakóhelyünkön, Kecskeméten ugyan rendelkezésre áll egy online menetrend, de útvonalat nem tud tervezni, ahogyan a Google Maps sem. Erre a problémára nyújt segítséget a Menetrend alkalmazás.

### 2. Probléma megoldásának menete

Egy város összes megállóját és járatát figyelembe véve megtalálni a leggyorsabb útvonalat egy embernek már komoly feladatot jelent. Szerencsére a megfelelő eljárásokkal egy számítógép ezt gyorsan el tudja végezni. A probléma megoldásához a város buszhálózatát egy gráfnak tekintettük, melyben a megállók a gráf pontjai, és a megállók között közlekedő buszok a gráf élei.

#### 2.1 Az adatok rendszerezése

Mielőtt nekiláthattunk volna a legrövidebb útvonal kereséséhez, szükségünk volt egy rendszerezett adathalmazra. Először megterveztük, hogy az útvonalkereséshez milyen adatbázis táblákban kell eltárolni az információkat, majd megkerestük, hogy honnan tudnánk ezeket az adatokat beszerezni. Szerencsére a [menetrend.kecskemethu](http://menetrend.kecskemethu) webhelyről HTTP lekérdezésekkel le tudtuk tölteni az adatokat JSON formátumban, és néhány sor kóddal rendszereztük őket a megfelelő táblákba egy SQLite adatbázisban.

#### 2.2 Útvonal kereső algoritmus

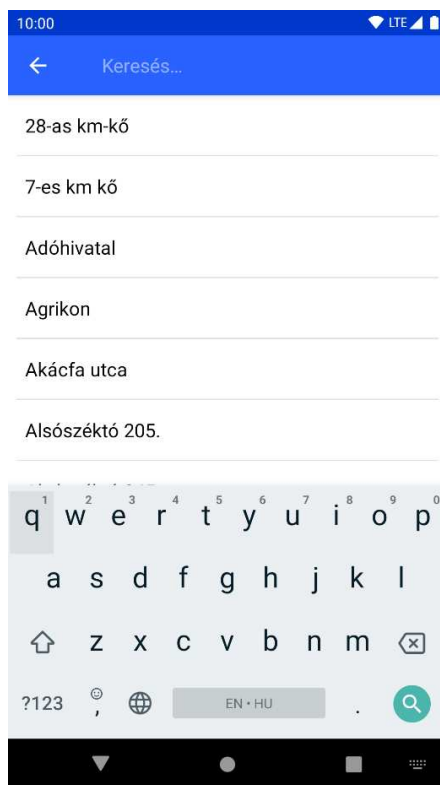
A buszhálózatot ábrázoló gráfban az élek irányítottak, mivel minden járat egy irányban közlekedik, és súlyozottak is. Két megálló közötti él súlya függ a menetidőtől és a várakozási időtől. Némi kutatómunka után megtaláltuk, hogy

melyik bejárési algoritmus lenne a leoptimálisabb ehhez a problémához: Dijkstra algoritmusára esett a választás. Az útvonaltervezés alapvetően erre épül, némi változtatásokkal. A súlyok meghatározásánál ugyan a menetidőt előre tudjuk, viszont a várakozási idő attól függ, hogy mikor ér a megállóba az előző járat, és mikor indul a következő. Így tehát minden él súlyát futásidőben számolja ki a program.

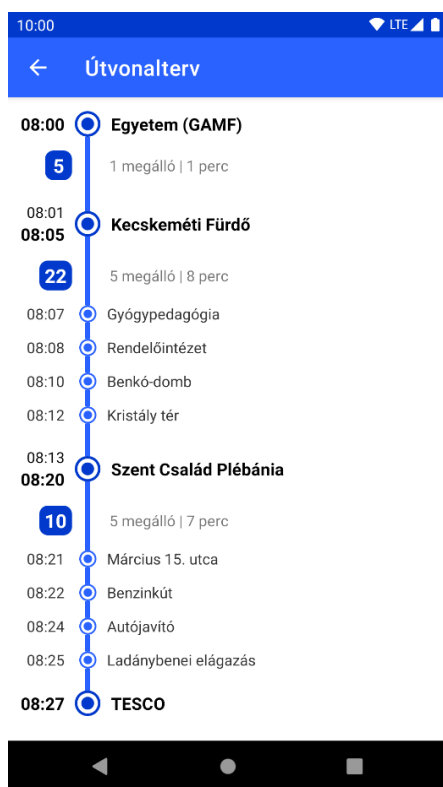
### 2.3 Alkalmazás, és felhasználói felület

Az algoritmus egyszerű, és a jövőben akár szélesebb körű felhasználását egy androidos alkalmazás teszi lehetővé. Az adatbázis az alkalmazásban van, így az offline használható, nem szükséges hozzá internet kapcsolat. A felület egyszerű és rendezett. A főképernyőn, ami az 1. ábrán látható, megadható a kiindulási és cél megálló, valamint az indulás időpontja. A megállót egy listából lehet kiválasztani, amit egy kereső mező könnyít meg (2. ábra). A megtervezett útvonal pedig a 3. ábrán látható.

1. ábra: Főképernyő



**2. ábra: Megálló választása**



3. ábra: Útvonalterv

### 3. Elért eredmények

Az alkalmazás erre a versenyre készült, máshol még nem volt bemutatva. Úgy gondoljuk, hogy egy ilyen jellegű program sokak mindennapi közlekedését könnyítheti meg. Habár eddig csak a kecskeméti menetrend érhető el, és a közel jövőben az alkalmazás funkcióinak bővítése és tökéletesítése a célunk, egy hasonlóképpen rendszerezett adathalmazon ugyanúgy működik az útvonaltervezés. Ez lehetővé teszi, hogy a későbbiekben akár más városok menetrendjét is kezelni tudja a program.