

Rászoruló embertársaink megsegítése szociális média felületeken

Raven Sztupák Szilárd Zsolt

Eötvös József Collegium*

zsolt@raven.scot

1. Bevezetés

Az egyetem elvégzése után számos lehetőség adódik a volt hallgatónak, hogy mit kezdjenek életükkel. Vannak, akik a szakmán kívül helyezkednek el, mások maradnak az egyetem csábító közelségében. Elég gyakori az is, hogy a korporét élet GDP termelési mutatói veszik le lábukról az embereket, akik emiatt úgy döntenek, hogy a magánszektorban folytatják pályafutásukat. Függetlenül attól, hogy milyen életpályát választunk, a megszerzett tudásunkkal ne legyünk önzők, és ahol tudunk, próbáljunk segíteni felebarátainkon.

Vannak erre kifejezetten szakosodott nemzetközi közhasznú szervezetek. Egyik ilyen a CodeBar[1], ami a technológiai szektorban alul-reprezentált embertársainknak segít azzal, hogy keres nekik tapasztaltabb mentorokat a problémáik megoldásában. Sokáig volt Budapesten is aktív közösségük, azonban ez sajnos 2019 körül önkéntesek hiányában elapadt. Segítő szándékú fejlesztők jelentkezését nemzetközi szinten azonban szívesen várják, akár online, távmunkában is lehet nekik havonta 1-2 óra ráfordítással besegíteni. A szerző emellett természetesen nagyon reméli, hogy idővel a budapesti közösség is újra aktív lesz.

A másik hasonló nemzetközi kezdeményezés pedig a MigraCode[2], mely menekülteknek, és hátrányos szociális helyzetből származó embereket segít programozni tanítani. Nekik sajnos nincs kirendeltségük Magyarországon, de több országban aktív szervezetük is szívesen várja a távmunkában segíteni szándékozó jelölteket!

*2004–2009

Azonban nem ez az egyetlen módja a segédkezésnek. Ha valaki szemfüles, és odafigyel, akkor szociális média felületeken is könnyedén találhat oltalomra szoruló illetőket. Az alábbi cikkben három példán keresztül mutatom be hogyan tudunk embertársaink támogatására sietni, ha szükségük lenne rá!

2. Kossuth Lajos azt üzente

sztivan felhasználónevű Tumblr felhasználó egy hűvös februári napon úgy érezte, hogy egy fontos információt szeretne a világról megtudni. Ez az információ pedig nem más, mint amit a 1. ábrán ábrázolt képernyőmentés ábrázol[3].

az irreleváns kérdés, amire megpróbálnám keresni a választ, ha végtelenül sok fölösleges időm volna

hol található Magyarországon a hely, ami a legtávolabb van egy Kossuth utcától?

1. ábra. Problémafelvetés

Ez pontosan az a fajta kérdés ahol a több évnyi Eötvös Collegiumban szerzett tapasztalatomat alkalmazni tudom. Aktív szociális életet élő lényként a Földrajz-Földtudomány műhely tagjaival gyakran szerveztünk közös programokat, ami során megismerkedtünk egyik kedvenc¹ foglalkozásukkal, ami nem más, mint térképek digitalizálása. Ez a nagyszintű precizitást és monotontúrét igénylő feladat lehetővé teszi, hogy régi, papír alapú térképeken szereplő adatokat digitális eszközökkel kezeljünk, rajtuk különféle lekérdezéseket futtassunk le. Ezúton is szeretném köszönetemet kifejezni azokkal szemben, akik ezeket a tevékenységeket önrőből, gyakran anyagi kompenzáció nélkül elvégezték, és végül munkájuk eredményét ingyen elérhetővé tették a világhálón, főleg mert így ezt már nem nekem kell megtenni.

Nemzetközi viszonylatban a legismertebb ilyen digitalizációs projekt az EC Informatikai Műhelyével egyidős OpenStreetMap[4], mely célja egy egész bolygót lefedő, ingyenes térképészeti adatbázis létrehozása. Hasonló projekt a szintén 2004 óta létező magyar Turistautak[5] is,

¹gy.k.: gyűlölt

mely Magyarország és a környező országok turistaútjainak feltérképezését tűzte ki céljául. A két projekt sokáig egymás konkurenciájaként üzemelt, többek között a különböző licenszfeltételeknek köszönhetően: míg a Turistautak térképének böngészése kizárólag non-profit céllal volt engedélyezett, addig az OpenStreetMap adatait kereskedelmi célokra is fel lehetett használni. 2015-ben történt egy változás, amikor a Turistautak szerkesztősége szintén teljesen szabaddá tette az adatokat, ami után a két projekt adatbázisa összefésülésre került. Ennek eredményeképpen az OpenStreetMap Magyarországgal kapcsolatos része is élvezhette a több évnyi, önkéntes digitalizációs munka eredményét.

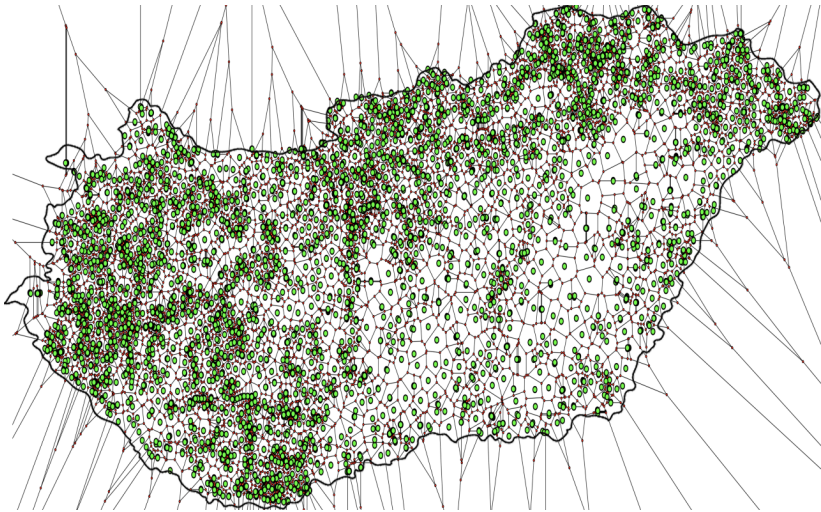
Ez nekünk pont kapóra jön, mert a kérdés megválaszolásához szükségünk lesz Magyarország összes Kossuth utcájának a listájára, amit az OSM adatbázisából fogjuk beszerezni. Számos eszköz létezik, hogy az OSM adatait saját számítógépünkön kezeljük, az `osmosis`, valamint az `osmconvert` parancssoros, míg például a `QGis` grafikus feldolgozást tesz lehetővé. Mindegyik említett eszköz ingyenes, nyílt forráskódú. A 2. ábrán látható módon fogjuk őket használni, először, hogy az egész Magyarországot tartalmazó adatsort leszűrjük csak a nevesített utakra (amelyek az OSM rendszerben `highway` jelzéssel vannak ellátva), majd utána hogy ebből a szűrt adatokból egy CSV állományt nyerjünk ki a Kossuth utcák középpontjának GPS koordinátaival.

```
$ osmosis -read-pbf-fast hungary.osm.pbf
    file="hungary.osm.pbf" -way-key keyList=highway
    -way-key keyList=name -used-node -tag-filter
    reject-relations -write-xml file="hungary-roads.osm"
$ osmconvert hungary-roads.osm -all-to-nodes
    -csv="@id @lon @lat name" -csv-headline | grep -i
    kossuth > streets.csv
```

2. ábra. Kossuth utcák adatainak kinyerése az OSM adatbázisából

Amint kinyertük az utcák koordinátáit, keresnünk kell egy algoritmust, ami segít nekünk eldönteni, hogy hol találjuk meg a megoldást. Egy gyors Google használat sejteti, hogy nagy valószínűséggel egy Voronoi-diagramot érdemes készíteni az utcák adatainak alapján[6]. A Voronoi-diagram ugyanis a teret úgynevezett Voronoi-cellákra osztja, ahol minden egyes kiinduló elemhez (a mintánkban minden egyes Kossuth utcához) pontosan egy cella tartozik. A cellák jellegetessége,

hogy a azt a területet határolják körbe amin belül minden egyes pont a cellához rendelt elemhez (azaz Kossuth utcához) közelebb van, mint akármelyik másik elemhez (Kossuth utcához). Ebből következik, hogy a cellák határain vannak azok a pontok, amelyek egyenlő távolságra vannak több Kossuth utcától is. Továbbgondolva: ezeken a határvonalakon lesznek a Kossuth utcáktól található legközelebbi pontok, ezen területen kell keresnünk majd azt a pontot, ami a szomszédos Kossuth utcáktól a lehető legközelebb van. Külön öröm az is, hogy a Voronoi-diagramok általában nagyon szépen néznek ki, feldobva ezzel a róluk szóló cikkek élvezeti értékét.



3. ábra. Kossuth utcák Voronoi-diagramja Magyarország felett

A 3. ábrán látható térképet nézegetve egyből szembetűnik, hogy a Tiszántúlon a ritkább népsűrűséghez ritkább Kossuth-koncentráció is tartozik. Naívan így azt gondolhatjuk, hogy az eredményt is majd ott kell keresni. Ha a fent leírt algoritmust lefuttatjuk, ténylegesen ott is fog nekünk eredményt találni, mégpedig a Hortobágyi Nemzeti Park környékén². Ez a pont azonban nem a legközelebbi pont, mivel algorit-

²Az algoritmus első futtatásakor Körösladány környékére tette az eredményt, azonban alapos vizsgálat kimutatta, hogy a város vasútállomásának bekötőútját a közelmúltban átnevezték Vasút utcáról Kossuth-ra, mely átnevezésről az OpenStreetMap nem tudott. Szerencsére, mivel az OSM adatbázisa bárki által szerkeszthető, ez a hiba azóta már orvosolva lett a szerző által

musunkba egy hiba csúszott.

Első éves hallgatóként anno megtanultuk, hogy az egyik leggyakoribb programozási hiba nem gondolni arra, mit csinál a programunk, vagy algoritmusunk az adatsorok határán. Bár általában ezt képletesen kell érteni (tömbök esetén például a tömb elején és végén érdemes megvizsgálni jól működik-e a programunk), itt azonban tényleges határról van szó: ha ügyesek vagyunk észrevevesszük, hogy nemcsak a Tiszántúlon, hanem a határmenti területeken is ritkábbak a Kossuth utcák. Itt azonban algoritmusunk nem működik kielégítően, hisz az csak a Voronoi-cellák metszéspontját vizsgálta eredetileg. A hiba javításához a határmenti cellákra rá kell illesztenünk Magyarország határvonalát, és a kettő metszésében található pontokat is meg kell vizsgálnunk.

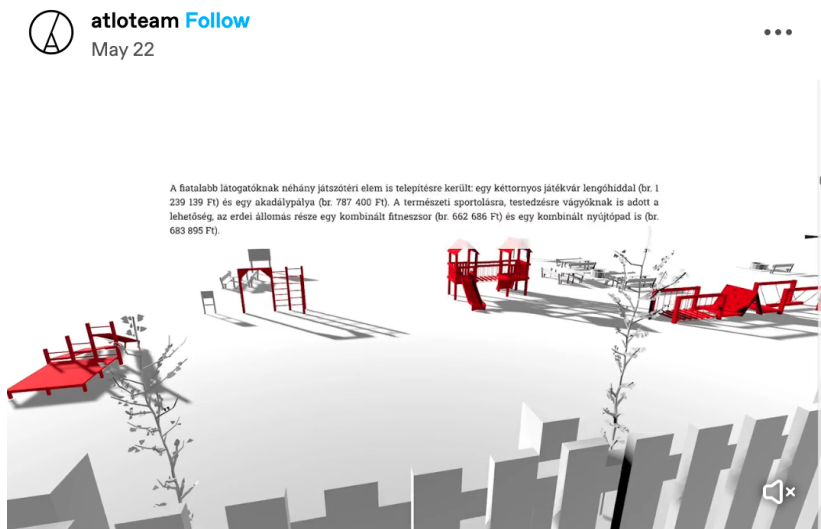
Az algoritmust újra lefuttatva megkapjuk a tényleges nyertest: a szlovén-osztrák-magyar hármashatár pontot. Innen a legközelebbi Kossuth utca úgy 17km-re található, Szentgotthárdon. Ha egy túra során el szeretnénk zárandokolni ide, a Kossuth utcától való félelmünkben, akkor örülhetünk, ugyanis ez a lokáció több okból is híres. Egyrészt hármashatár pont, másrészt ez Magyarország legnyugatibb koordinátája is. A helyszínen így egy kisebb pihenőt, játszóteret, valamint piknikezésre alkalmas padokat is találunk, hogy teljes mértékben kiélvezhessük túránk gyümölcsét. Azonban, ha egy kicsit nagyobb kihívásra vágyunk, a második helyen a Hortobágyi Nemzeti Park területén található ponton leszünk legmesszebb a Kossuth utcától, úgy 13km-re. Itt azonban nem lesz semmi indikátora a helyszín fontosságának, mindenképp érdemes egy jól feltöltött, és precíz GPS lokátorra hagyatkoznunk a célpont megtalálásához.

A teljes adatsor, valamint egyéb példák, mint például Petőfik, Rákóczyk vagy József Attilák megtekinthetők a projekt honlapján[7], ahol kiderül, hogy Magyarországon bárhol is vagyunk lesz 12km-en belül egy sportlétesítmény, az Egyesült Királyságban meg nagy átlagban – különösen Angliában – ha véletlenszerűen ledobnak egy repülőből³, akkor 1km-en belül fogunk találni egy kocsmát. Ami jól jön ha esetleg egy kis frissítőre vágynánk a landolás után.

³remélhetőleg működő ejtőernyővel

3. A lombkoronák réme

2023. év elején nagy port kavart egy hírhedt falétesítmény az ország keleti felén[8]. Számos hírportál írt cikket, és vizsgálta az ügyet. Az Átlátszó adatvizualizációs csapata, az ATLO Team is készített a létesítményről egy infografikát, melyet a 4. ábrán látható felhívással tettek közzé[9].



Link: <https://atlo.team/wp-content/uploads/2023/05/lombkorona.html>



felmerült, hogy lehetne belőle egy fps-t csinálni. innen le lehet tölteni a 3D-modellt :) <https://github.com/szabokrissz96/lombkorona>

4. ábra. A problémafelvetés

Természetesen kaptam az alkalmon, hogy megnézzem mit is lehet

egy ilyen modellel tenni! 3D grafikával volt némi kapcsolatom már középiskolai éveim során is, amikor megjelentek a korabeli webes 3D rendszerek, például a VRML. 2D alapú játékefejlesztés mindig is érdekelt, több 2D alkalmazás fejlesztésében is részt vettem, de 3D grafikával nem túl sokat foglalkoztam. Később, egyetemi éveim alatt bár jártam 3D grafikával foglalkozó extra kreditre, de szerintem csak azért sikerült azt teljesítenem, mert a vizsgán az első tételt húztam, ami bevezetésekképpen csak két dimenziós terekkel foglalkozott.

Mindezt összevetve kíváncsivá tett mennyit fejlődött a technológia az elmúlt 25-30 évben, azóta amióta először találkoztam a VRML nagyon korai példányaival. Úgy rémlik már akkor is sikerült benne egy nagyon-nagyon kezdetleges FPS-t⁴ elkészítenem, bár sajnos már nincs ennek nyoma az archívumomban. Persze nem kell túl sokat gondolni: egy lapos szürke sík terep volt a talaj, és pár kockadarab jelezte a falakat rajta. Ellenfél nem volt, csak statikus gömbök (talán sárga színűek), amelyeket fel kellett vened. A falak pedig azt sem érzékelték ha neki mentél, simán át lehetett siklani rajtuk. Persze ne feledjük el, ez még bőven a múlt évezred végén volt, amikor még az Internet Explorer és a Netscape Navigator volt a két legismertebb böngésző, és sokan gondolták, hogy a honlapokon majd Java alapú alkalmazások fognak futni. Szerencsére azóta már se Internet Explorer, se Netscape Navigator, és a Java alapú alkalmazásokat is csak az APEH – mai nevén NAV – vette komolyan.

No de lássuk mennyi mindent fejlődött a világ úgy 25 év alatt: egy kis Google keresés kidobott egy Three.JS keretrendszer alatt írt FPS rendszert[10], amiben csak a terepet kellett lecserélni a nyírmártonfalvi lombkoronasétány modelljére, és 5 perccel később már kész is volt egy nagyon kezdetleges, de működő játék. Lehetett benne mozogni, löszert felvenni a földről, lövöldözni, a tereptárgyaknak neki lehetett menni (és nem siklottál át rajtuk), valamint az ellenfél is próbált némi intelligenciát mutatni azzal, hogy megpróbált üldözni, meg csapkodni. Bár természetesen a grafika nem egy AAA kategóriás játékot idézett, lényegesen jobban nézett ki, mint bármi más, amit pár perc alatt, webes platformra össze lehetett hozni, mondjuk úgy 10 éve.

Az igazán érdekes rész viszont csak ezután következett. Bár egy valamennyire működő játék 5 perc alatt elkészült, azért ez nem volt

⁴Belső nézetű lövöldözős játékot

túl nagy teljesítmény, tekintve hogy csak más emberek által ekészített projekteket hoztam össze egy kalap alá. A Telex, akinek tudomására jutott a honlapom is elintézte az egész próbálkozást azzal, hogy „(csak) arról van szó, hogy valaki ráhúzta egy már létező kezdetleges, belső nézetes demóra a lombkoronasétány virtuális mását”[11]. Így írjon rólad szépeket a magyar média ugyebár. Pedig ők már a javított változatot látták, mert hát a projekt nem állt le, sőt most kezdett csak érdekessé válni az egész.

A rendszerben még volt bőven hiba, és mivel a játékmotort nem magam írtam, ezért ezek kijavításához először rá kellett jönni, hogy mi, hogyan, és miért úgy működik, ahogy. Szerencsére az eredeti kód nem volt túl komplex (bár nem is volt túldokumentálva), így sikerült hamar megfejteni a működésének alapjait, és a vizsgálgatás során lehetett tanulni egyet s mást a 3D grafika, valamint a játékfejlesztés alapjairól. A két nap, amit ebbe a projektbe fektettem, mindenképp többet tanított nekem, mint egy fél évnyi 3D grafika az egyetemen. Vagy lehet csak izgalmasabb volt!

Néhány példa, hogy miket sikerült ez alatt a pár napos fejlesztés alatt tanulnom:

- A navmesh működése – 3D modell ami megmutatja az AI által irányított lényeknek, hogy hova mehetnek, és hogyan jutnak oda a leggyorsabban
- Különféle textúrák működése webes környezetben, beleértve az égbolt és talaj rendes mintázatát
- Különféle ütközésvizsgálati módszerek, statikus-dinamikus illetve dinamikus-dinamikus modellek között
- Játéktechnikai javítások – jobb AI kezelés, különféle nehézségi szintek, többfajta sebezési pontok támogatása
- Webes hangkezelés 3D térben, valamint a játék kezelésének támogatása mobil környezetben

Az elkészült projekt és a forráskód hozzá megtekinthető a szerző honlapján[12].

4. Másfélmillió lépés Magyarországon, tömegközlekedve

A 3D-s kitekintő után egy újabb térképészeti feladat az 5. ábrán, ezúton `mindenfoglaltmar` nevű felhasználótól[15].

Van-e valakinek ötlete olyan kék túra szakaszra, ami 15-20 km közötti és megközelíthető úgy, hogy A ponton hagyom a kocsit, eltömegközlekedek B pontra és visszasétálok A pontba.

5. ábra. Problémafelvetés

Megint egy tökéletes feladat, ha esetleg túl sok határidős munkánk lenne, és egy kis agyi felfrissülésre vágynánk – miközben természetesen segítünk valakinek megoldani világraszóló problémáját. A fenti feladat hasonlóan térképészeti megoldást igényel, mint a Kossuth térképés, azonban más forrásokból kell beszerezni hozzá az adatokat. Az ötlet az az, hogy megpróbáljuk megtalálni az összes létező kombinációját az olyan túráknak, amelyek egy buszmegállóból indulnak, eljutnak egy másik buszmegállóba tömegközlekedve, onnan egy rövid sétával továbbvisznek egy kéktúrás pecsételőhelyhez, majd onnan elvezetnek a kiszemelt Kéktúra szakaszon keresztül vissza abba a buszmegállóba, ahonnan eredetileg is indultunk. Nézzük honnan szerezzük be az adatokat ehhez.

- Szükségünk lesz a Kéktúrák teljes útvonalára. Ez elérhető a Kéktúra hivatalos honlapjáról GPX formátumban. Az egyszerűség kedvéért a három különböző Kéktúra vonalát **QGis** segítségével kézzel összefűztem egyetlen Magyarországot átszelő Kékkörre
- Szükségünk lesz Magyarország összes buszmenetrendjére – vagy legalábbis azokra, melyek Kéktúra szakaszok közelében helyezkednek el. Mind a BKK, mint a Volánbusz ezt az adatsort ingyenesen elérhetővé teszi bárki számára a honlapján, GTFS formátumban. A MÁV oldalán sajnos ez nem szerepel, de ezt nem tekintjük túl nagy problémának. A Budapest környéki vasútvonalak, például a HÉV menetrendek így is szerepelni fognak a rendszerben, a BKK adatbázisa miatt.
- Végezetül turistautat kell találnunk a buszmegállók és a Kéktúra szakaszok kiinduló és végpontjai között. Erre megint az OSM

adatbázisát fogjuk használni, onnan nyerjük ki a potenciális gyalutakat.

Miután a nyers adatokat beszereztük, fel kell őket dolgozni. Először is minden, a Kéktúra szakaszok végpontjaihoz közel lévő buszmegállót összegyűjtünk. Az egyszerűség kedvéért az egymástól csak párszáz méterre lévő buszmegállót egynek tekintjük, mivel általában ezek az út két szemközti oldalán helyezkednek el. Miután kiszűrtük a buszmegállókat, lefuttatunk egy keresést, ami megkeresi a legrövidebb turistautat a buszmegállók és a Kéktúra szakaszok végpontjai között. A szerző több különféle technológiát is kipróbált, végül az adatokat egy PostGIS[13] adatbázisba töltötte be és a `pg_routing` programcsomag segítségével kereste meg az eredményt, A* algoritmus segítségével.

A következő lépcsőfok a buszútvonalak kiderítése. A GTFS adatsort egy OpenTripPlanner[14] nevű programba töltjük be, ahonnan ezután lekérdezhetjük, hogy A és B pont között milyen útvonalak léteznek. Ezt a keresést szintén lefuttatjuk az összes buszmegállópár között (bizonyos távolsági kereteken belül), majd az eredményeket sorbarendezzük kritériumaink alapján, hogy megkapjuk az optimális buszmenetrendeket. Ilyen kritériumok például, hogy melyik a leggyorsabb, a legsűrűbb, vagy esetleg a legkevesebb átszállást, illetve gyaloglást igénylő útvonal.

Miután minden adat megvan már csak egy honlapot kell fejleszteni, ahol a három különböző szekciót egy térképen ábrázolunk, kereshető formátumban. Persze az eredeti kérdés csak a 15-20km-es szakaszokra vonatkozott, ha igazán hasznos honlapot szeretnénk készíteni gondolkunk olyanokra is, akik ennél csak kevesebbet, vagy ellenkezőleg jóval többet szeretnének túrázni.

Mindenesetre az eredeti kérdésre a válasz: rengeteg ilyen útvonal létezik, Budapest környékén egyből három ilyen szakasz is van: A Dobogókő – Kevély nyereg, a Piliscsaba – Dorog, valamint a Mogyorósbánya – Dorog szakasz mind bejárható egy nap alatt egy rövid kis 15-20km sétálással, de pár szakaszt leszámítva majdnem az egész Kékkör elvégezhető napi 20-25km sétálással úgy, hogy mindig visszajutunk a kiindulópontunkra a nap végén.

A Kékkör kereső alkalmazás és forráskódja szintén megtalálható a szerző honlapján[16].



6. ábra. Dobogókő - Kevély nyereg túraajavaslát

5. Utószó

Remélem a fenti pár példa meghozta az olvasók kedvét arra, hogy érdekes problémafelvetések megoldására új technológiákat próbáljanak ki, különösen ha azok amúgy nem tartoznak a nap-mint-nap használt eszköztárukba. Még ha nem is sikerül megoldani a problémát, rengeteget lehet tanulni a sikertelen kísérletekből is. És természetesen ne feledjünk, hogy nem csak furcsa kérésekkel lehet másokon segíteni, a bevezetőben említett szervezetek folyamatosan várják önkéntesek jelentkezését. Tanulni vágyó illetők mindig lesznek, és mindig lesz szükség arra, hogy valaki tanítsa őket. Ötleteket, új technológiákat, érdekes problémákat ezen alkalmak során is bőven lehet gyűjteni.

Ha sokáig csináljuk ezeket a projekteket, előbb-utóbb utolér a hírnév is, így könnyen lehet, hogy amikor valaki megkérdezi tőlünk, hogy „te, meg tudod mondani, hogy melyik az a busz, ami a legtöbb Kossuth utcán halad át”, és akkor erre már tudod is egyből a választ, hogy „természetesen, amit te keresel az a Nagykanizsa és Zalaegerszeg között közlekedő 6458-as busz, ami rekordmennyiségű, hét különböző Kossuth utcán is megáll, az egyiken kétszer is, a két végén”.

„Hétköznap érdemes felszállni rá reggel 6:25-kor, Nagykanizsán.”

A szerzőnek számos más projektje is van, különösen szeret térképészeti problémákat megoldani és politikai indíttatású játékokat készíteni. Portfóliója megtalálható a honlapján[17].

Hivatkozások

- [1] Codebar is a charity that facilitates the growth of a diverse tech community by running free regular programming workshops for minority groups in tech. codebar.io
- [2] MigraCode Europe A European Network to promote Open Tech Education for Refugees and Migrants. migracode.eu
- [3] Az észtek litvánok lettek, ID #182822065158 sztivan.tumblr.com/post/182822065158
- [4] OpenStreetMap, a free, open geographic database updated and maintained by a community of volunteers via open collaboration. openstreetmap.org

- [5] turistautak.hu Magyarország leggyorsabban fejlődő nonprofit térkép-portálja turistautak.hu
- [6] Wikipédia-szerkesztők, 'Voronoi-cella', Wikipédia, 2022. május 22. hu.wikipedia.org/w/index.php?title=Voronoi-cella
- [7] Zsolt Sz Sz Raven, Determine furthest points away in a country for a specific set of points sztupy.hu/kossuth-map
- [8] Sokszínű Vidék, Kivágott erdő helyén épült lombkorona-sétány Nyírmártonfalván sokszinuvidek.24.hu/mozaik/2023/03/23/lombkorona-~setany-nyirmartonfalva-nyirseghajdu-bihar-megye-erdo
- [9] ATLO Team, powered by ATLATSZO. ID #718041277567025152 atloteam.tumblr.com/post/718041277567025152
- [10] Mohsen Heydari, Three FPS Demo github.com/mohsenheydari/three-fps
- [11] Flachner Balázs, Virtuálisan ugyan, de már mutánsokra is lehet lövöldözni a lomb nélküli lombkoronasétánynál telex.hu/zacc/2023/05/24/lombkoronasetany-~nyirmartonfalva-videojatek-lovoldozes
- [12] Zsolt Sz Sz Raven, Canopy FPS sztupy.hu/lombkorona
- [13] PostGIS extends the capabilities of the PostgreSQL relational database by adding support storing, indexing and querying geographic data. postgis.net
- [14] OpenTripPlanner (OTP) is a family of open source software projects that provide passenger information and transportation network analysis services. opentripplanner.org
- [15] SztupY, ID #728193481495003136 tumblr.sztupy.hu/post/728193481495003136
- [16] Zsolt Sz Sz Raven, Kékkör szakasz kereső sztupy.hu/kekkor-kereso
- [17] Zsolt Sz Sz Raven, Projects page sztupy.hu/projects