



Online feladat
2019



SZÉCHENYI  2020



MAGYARORSZÁG
KORMÁNYA

Európai Unió
Európai Szociális
Alap



BEFEKTETÉS A JÖVŐBE

BeeSmarter 2019

A verseny előfordulójában egyszerű Android játékot kell készíteni, ami telefonokban található mozgásszenzorok méréseit használja fel mozgáskoordinációs játék elkészítésére.

A játék működése

A játék a következő egyszerű lépésekből áll:

- Az alkalmazás elindítását követően legyen lehetőség kiválasztani a játék módját
 - Két játékos
 - Több játékos
- A játékmód kiválasztását követően a játszandó feladat kiválasztása következik
 - Telefon megrázása a lehető legjobban (ügyeljenek, hogy ne dobjátok el a készüléket), pár másodpercig.
 - Telefon megrázása egy megadott ritmusban (frekvencián) 10 másodpercen keresztül.
 - Egyszerű forma rajzolása a telefonnal a levegőben (például háromszög), megadott idő alatt.
 - Extra pontért további ötletek
- A beállításokat követően elindítható a játék. Először az első játékos hívja, majd kiírja, hogy milyen játékmódban van.
- Ezt követően indítható a játék, egy Start gomb érintésével, amelyet visszaszámlálás követ.
- A visszaszámlálást követően kell végrehajtani a feladatot, ami alatt a méréseket végzi az alkalmazás, a feladathoz tartozó ideig. (Ezt szabadon meghatározhatjátok.)
- A mérést követően ki kell értékelni, hogy mennyire sikerült a feladat végrehajtása.
 - Rázás esetén a szenzorok által mért legnagyobb gyorsulás értéke.
 - Ritmus esetén a ritmus pontossága.
 - Forma esetén meg kell becsülni, hogy mennyire sikerült a feladat kivitelezése, például a háromszög három mozgási fázisból áll-e?
 - Stb.
- A kiértékelésre számíthatok ki egy pontértéket, amely pontérték alapján lehet eldönteni, hogy melyik játékos teljesítette a feladatot.
- A pontérték számítását követően az applikáció hívja a következő játékost, vagy ha nincs több játékos, akkor a kezdőképernyőre tér vissza.

Továbbfejlesztési ötletek:

- Egyjátékos mód, amikor a feladat az, hogy minél pontosabban ismétlje el az egyetlen játékos a mozdulatot, többször. (És ezt lehet értékelni.)

- A mozgás képernyőn történő kirajzolása, vagy valamilyen vizualizáció készítése.
- Mozgásminták tanítása az alkalmazásnak, ami játékeladat lehet.
- A több játékos mód egyszerre játszása több telefonon – WiFi/Bluetooth kommunikációval.
- A feladat végrehajtásának automatikus felismerése – tehát nem időre megy, hanem a telefon észreveszi, ha a játékos abbahagyta.

Segítség

Az android telefonokban található egy háromtengelyű gyorsulásmérő, valamint giroszkóp. A gyorsulásmérővel a tengelyek mentén fellépő gyorsulás pillanatnyi értékét lehet mérni m/s^2 -ben, a giroszkóp pedig szöggyorsulást mér. Android dokumentáció [itt](#) és [itt](#).

Fontos, hogy a rögzített adatoktól függően (ACCELEROMETER, LINEAR_ACCELERATION GRAVITY) a nehézségi erő okozta gyorsulásérték hozzáadódhat a mért értékhez. Keressétek meg, hogy melyiket érdemes használni.

A jel feldolgozása során érdemes lehet vizsgálni az eredő gyorsulást.

Noha egyértelmű, hogy ha ismerjük két pillanatnyi gyorsulásérték és a kettő között eltelt idő mértékét, akkor kaphatunk belőle egy pillanatnyi sebességértéket (és hasonlóan elmozdulást), sajnos az összegződő számítási hiba nagyságrendje azonos lehet a számított értékével, így nem pontos az eredmény. (Profiknak: kétszer integrálunk idő szerint, úgy hogy az integrálás során figyelembe veendő konstans értékét nem ismerjük. Emiatt jobb híján nullának tekintjük azt, ami viszont okozza ezt a hibát.)

A mért jel frekvencia-összetevőit ki lehet nyerni, úgynevezett, Fourier transzformációval, pontosabban jelen esetben diszkrét Fourier transzformációval. (DFT, illetve [FFT](#)). Ennek az elkészítése és felhasználása extra pontokat jelent a feladat kiértékelése során.

A verseny szervezőit a beesmarter@itk.ppke.hu címen lehet elérni, illetve a forduló alatt nyitva tartunk egy [Slack](#) csatornát, ahol a közérdekű információkat tesszük közzé, illetve kérdéseiteket is itt tehetitek fel.

Milyen nyelv és milyen libek?

Elsősorban Java, illetve Kotlin. Ha ragaszkodtok hozzá, akkor C++ natív libek is. Külső programkönyvtárak használata engedélyezett, a megfelelő licenzfeltételek betartása mellett. (Amennyiben egy csapat ezt nem tartja be, akkor automatikusa kizárjuk a versenyből.)

Feladat beadása

A verseny weboldalán (<https://beesmarter.itk.ppke.hu>) bejelentkezést követően fel lehet tölteni a megoldásokat. A határidő április 24., szerda, 23:59.

Mit kell beadni és mit nem?

Az alkalmazás forráskódját és a kész alkalmazást kell beküldeni egy tömörített zip fájlban.

A projekt mappában található fájlok és mappák közül ne adjátok be

- a build mappákat (build/*, app/build/*)
- és az android-studio beállítás fájljait (.idea/*).
- Nincs szükségünk az egyéb helyi információkat tároló fájlokra sem (pl. local.properties, [több információ](#)).

Mindenképpen be kell adni:

- Az összes fájlt, ami a projekt lefordításához szükséges:
 - a forráskódokat és az xml fájlokat,
 - a build.gradle fájlokat,
 - a külső library-eket, ha használtatok.
- Egy .apk fájl is, ami az alkalmazásod legfrissebb verzióját tartalmazza.

Emellett kérünk beadni egy rövid leírást, ahol az elkészített alkalmazást mutatjátok be. (Ebben a dokumentumban külön bemutathattok további, más projekteket is, amelyen dolgoztatok – ezt pontegyenlőség esetén figyelembe vesszük.)

Tesztkörnyezet

A beadott játékokat elsősorban Android 8.1 rendszerrel rendelkező Google Nexus 5X, illetve 9.0 rendszerű Google Pixel 2 XL telefonokkal fogjuk tesztelni, de a fejlesztés során ti Android 5.0 feletti bármilyen készüléssel dolgozhattok, ez nem fogja befolyásolni az értékelést. Azonban, ha valamiért úgy gondoljátok, hogy az általunk a kiértékelésnél használt telefonon az alkalmazásotok várhatóan nem fog jól működni (pl. csak emulátorban, tableten vagy régebbi telefonon tudtátok kipróbálni), akkor azt kérjük jelezzétek egy README.txt-ben a beküldött app mellett, a teszthez használt pontos környezet leírásával együtt.

Értékelési szempontok

A beadott megoldásokat a következő szempontok figyelembevételével rangsoroljuk:

- Az egyes megvalósított funkciók működőképessége, játszhatósága. (Elsősorban arra törekedjétek, hogy amit elkészítetek a működjön teljesen, tehát inkább kevesebbet, de jól.)
- A feladat kiírásban megadott funkciók megvalósítása.
- Alkalmazás stabilitása, hibakezelése.
- A forráskód, illetve algoritmusok struktúrája, a beadott program forráskódjának olvashatósága – „szépsége”.
- Grafikus felület felépítése, használhatósága.
- Extra pontok további ötletek megvalósításáért.

- A csatolt dokumentum az applikáció leírásáról, valamint más projektjeitek bemutatásáról.

A fenti szempontok alapján a rész megoldásokat is kiértékeljük, így érdemes beadni bármilyen állapotban levő megoldást is. A kialakult sorrend alapján határozzuk meg, hogy a verseny második fordulójába ki jut be.