

دانشگاه تهران
دانشکده‌ی مهندسی برق و کامپیوتر



سیستم‌های نهفته‌ی بی‌درنگ

تمرین سوم

آشنایی با سیستم عامل اندروید و استفاده از سنسور های تلفن همراه

طراحان:

روح الله ابوالحسنی (roohi.abol@gmail.com)

عماد جبار (emad.jabbarnk@gmail.com)

اساتید:

دکتر مهدی کارگهی، دکتر مهدی مدرسی

نیمسال دوم ۹۹-۱۳۹۸

1. مقدمه

در این تمرین قرار است با سیستم عامل اندروید و امکاناتی که جهت تعامل با سنسور های موبایل (حتما این لینک رو بخونید) در اختیار ما قرار می دهد آشنا شویم. همچنین در این تمرین با برخی محدودیت هایی که سیستم عامل جهت استفاده از سنسور ها و ارتباط با سخت افزار لایه پایین ایجاد می کند، آشنا خواهید شد.

امروزه تلفن های همراه پیشرفت چشم گیری کرده اند و دارای قدرت پردازشی بالا، تنوعی از سنسور های مختلف با دقت بالا و پشتیبانی از چندین شبکه ارتباطی مختلف مانند NFC ، WiFi ، Bluetooth و ... هستند. بعلاوه تمام این امکانات در ابعاد پایین و وزن کم در اختیار کاربران قرار گرفته اند.

حال سوال اساسی اینجاست که چطور می توان از این همه امکانات که همیشه همراه ماست استفاده بهینه کرد؟ آیا می توان از یک smart phone در کاربرد ها صنعتی و تجاری بجای برد های مرسوم استفاده نمود؟ چه محدودیت هایی برای این کار وجود دارد؟

در این تمرین پاسخی برای سوالات فوق پیدا خواهید کرد.

2. شرح تمرین

در این تمرین بازی Super Spinner Bros را که نسخه ای الهام گرفته شده از بازی Teeter Pro¹ است، پیاده سازی خواهید کرد.

در بازی Super Spinner Bros دو گوی وزن دار با دو جرم متفاوت (برای ساده تر شدن، گوی ها را لغزنده و نه غلطان فرض کنید) در دو نقطه در صفحه نمایش قرار می گیرند. ابتدا چهار مقدار زیر در برنامه وارد شده و سپس با زدن دکمه شروع بازی آغاز می شود.

- مکان اولیه گوی ۱
- مکان اولیه گوی ۲
- سرعت اولیه گوی ۱
- سرعت اولیه گوی ۲

¹ https://play.google.com/store/apps/details?id=pl.surix.teeterpro&hl=en_US

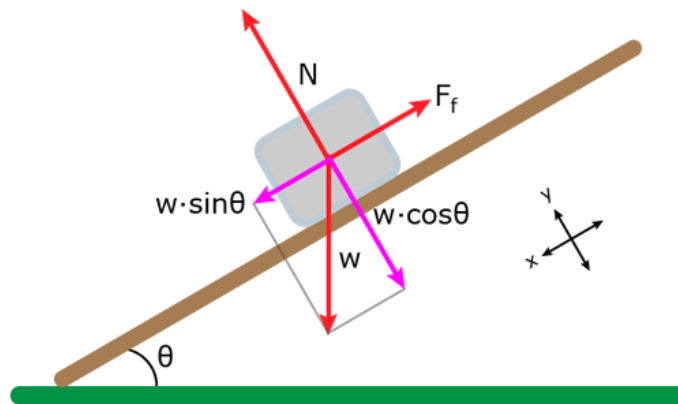
پس از آن گوی‌ها بر اساس زاویه تلفن همراه نسبت به افق و تغییرات زاویه و شتاب حرکت گوشی در جهت بردار سرعت محاسبه شده حرکت می کنند. دقت شود که در شرایطی ممکن است این دو گوی با یکدیگر برخورد کشسان داشته باشند که در این حالت‌ها، باید از قوانین پایستگی تکانه خطی و پایستگی انرژی برای محاسبه بردارهای سرعت پس از برخورد استفاده گردد. برای حرکت دادن گوی روی صفحه باید از قوانین فیزیک جهت محاسبه بردار سرعت و مکان در صفحه استفاده کنید. توجه داشته باشید که حرکت گوی‌ها باید پیوسته باشد و پرش تصویر نباید بوجود آید.

برای این کار نیاز است تا به کمک سنسورهایژیروسکوپ تعبیه شده در موبایل، زاویه موبایل را با محورهای سطح افق تشخیص دهید.

در پایان نیاز است تا با ابزار Systrace مجموعه eventهایی را که در سطح سیستم عامل رخ می دهد تا داده از سنسور خوانده شده و در صفحه نمایش گوی حرکت کند را رصد (profile) کنید و با توجه به نتایج آن، به سوالات بخش 6 در گزارش خود پاسخ دهید.

3. طراحی مفهومی

برای حرکت دادن گوی در صفحه نیاز است تا قوانین فیزیک حاکم بر مسئله تعیین گردد. در این تمرین جرم گوی اول ۱۰ گرم و جرم گوی دوم ۵۰ گرم است. گوی‌ها دارای قطر یکسان هستند. همچنین این دو گوی روی یک صفحه با ضرایب اصطکاک ایستایی $\mu_s = 0.15$ و دینامیک $\mu_k = 0.10$ قرار دارند. در شکل زیر، دیاگرام جسم آزاد برای یکی از این دو گوی نمایش داده شده است. دقت کنید که جسم رسم شده در این دیاگرام فقط در یک بعد حرکت می کند؛ حال آنکه گوی مسئله ما می تواند در دو بعد حرکت کند.



شکل 1

در شکل 1 زاویه θ در واقع یکی از زاویه های اندازه گرفته شده توسط سنسور ژيروسکوپ موبایل با صفحه افق است. همچنین F_f نیروی اصطکاک است. این نیرو در صورتیکه جسم در حرکت باشد نیروی اصطکاک دینامیک و در صورتیکه جسم در حال سکون باشد، نیروی اصطکاک ایستایی است.

در صورتیکه گوی در حال سکون باشد و اندازه برآیند نیروهای محرک جسم (ناشی از شیب) از مقدار نیروی اصطکاک ایستایی ($N \mu_s$) کمتر باشد، گوی حرکتی نمی کند. اما اگر مقدار این نیرو بیشتر باشد، گوی شروع به حرکت می کند و نیروی اصطکاک دینامیک به مقدار $N \mu_k$ و در خلاف جهت بردار سرعت گوی بر آن اعمال می شود.

همچنین اگر گوی در حال حرکت باشد، همواره نیروی اصطکاکی برابر با $N \mu_k$ در خلاف جهت بردار سرعت به آن اعمال می شود که m جرم گوی، g ثابت گرانشی (که در این مدل سازی مقدار آن $9/8$ در نظر گرفته شده است) و μ ثابت اصطکاک است.

نیروی N برابر با $mg \cos(\varphi)$ است که φ مساوی با زاویه بردار صفحه موبایل با محور z مرجع است.

در زیر بعضی از قوانین فیزیک مورد نیاز آمده است:

- رابطه نیرو و شتاب: $F = m a$
- رابطه شتاب و سرعت اولیه: $v = a t + v_0$
- رابطه شتاب زاویه ای و تغییرات زاویه: $\Delta \theta = \omega \Delta t$
- رابطه تغییر مکان یک جسم شتابدار با شتاب ثابت در یک بازه زمانی: $\Delta x = \frac{1}{2} a \Delta t^2 + v_0 \Delta t$

- رابطه پایستگی تکانه خطی برای دو جرم m_1 و m_2 در فضای یک بعدی:
- $$m_1 v_1 + m_2 v_2 = m_1 v'_1 + m_2 v'_2$$
- برای شبیه‌سازی برخورد دو گوی حتما [این لینک](#) را بدقت مطالعه کنید.

با توجه به اینکه در واقع شتاب اعمال شده به گوی‌ها ثابت نیست، بنابراین نیاز است تا بطور تقریبی حرکت گوی‌ها شبیه‌سازی شود. به این‌صورت که شتاب گوی در فواصل زمانی کوچک ثابت در نظر گرفته شود و محاسبات بردار سرعت و مکان بر اساس آن شتاب انجام شود. دقت کنید که حرکت گوی باید بصورت پیوسته احساس شود.

4. پیاده سازی

پیاده سازی این تمرین به دو صورت انجام خواهد شد:

- استفاده از سنسور Gyroscope

برای استفاده از سنسور Gyroscope می‌بایست سنسور مذکور را از `SensorManager` دریافت کنید. برای این کار می‌توانید از کد زیر کمک بگیرید:

```
private SensorManager sensorManager;
private Sensor sensor;
...
sensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
sensor = sensorManager.getDefaultSensor(Sensor.TYPE_GYROSCOPE);
```

برای دریافت داده از سنسور باید یک `SensorEventListener` در `SensorManager` ثبت نام² کنید.

برای این کار از متد `registerListener` در کلاس `SensorManager` استفاده نمایید. در شکل 2

می‌توانید اطلاعات مربوط به سنسور را مشاهده نمایید:

TYPE_GYROSCOPE	SensorEvent.values[0]	Rate of rotation around the x axis.	rad/s
	SensorEvent.values[1]	Rate of rotation around the y axis.	
	SensorEvent.values[2]	Rate of rotation around the z axis.	

شکل 2

- استفاده از سنسور Gravity

برای استفاده از سنسور Gravity در برنامه، مشابه سنسور Gyroscope عمل نمایید. به تفاوت خروجی این سنسور با سنسور Gyroscope توجه داشته باشید. می‌توانید اطلاعات مربوط به سنسور را در شکل 3 مشاهده نمایید:

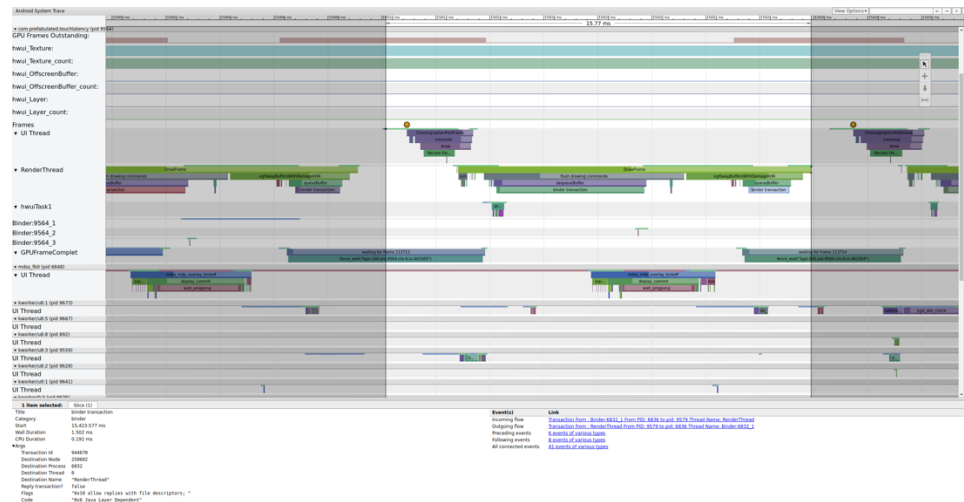
TYPE_GRAVITY	SensorEvent.values[0]	Force of gravity along the x axis.	m/s ²
	SensorEvent.values[1]	Force of gravity along the y axis.	
	SensorEvent.values[2]	Force of gravity along the z axis.	

شکل 3

برای تحویل این دو حالت، دو خروجی apk. تحویل خواهید داد. در صورتی که هر دو حالت را در یک فایل apk. در دو Activity مختلف پیاده سازی کرده و امکان سوییچ بین آنها به انتخاب کاربر وجود داشته باشد نمره امتیازی تعلق خواهد گرفت.

5. ابزار profile

ابزار Systrace نرم‌افزاریست که با استفاده از آن می‌توان اتفاقاتی را که در سیستم عامل رخ می‌دهد، رصد کرد. این برنامه روی سیستم شما اجرا می‌شود و موبایلی را که به سیستم متصل شده است profile می‌کند. در شکل 4 یک نما از این برنامه آمده است.



شکل 4

هر ریسسه^۳ در این برنامه یک نوار مختص به خود دارد که وضعیت آن ریسسه در زمان را نمایش می‌دهد. هر وضعیت یک رنگ خاص دارد. بطور مثال در مدت زمانی که یک ریسسه idle باشد، در نوار مربوطه به آن ریسسه، رنگ آن بخش خاکستری است. از این طریق می‌توان وضعیت زمانبندی ریسسه‌ها و پردازش‌های مختلف را مشاهده کرد.

برای نصب این برنامه ابتدا لازم است Android Studio روی ماشین شما نصب شده باشد. سپس:

- باید در **Android SDK > System Settings > Appearance & Behavior**، گزینه **Android Studio Tools** نصب شده باشد.
- Python باید نصب باشد و در **execution path** پروژه شما اضافه شده باشد.
- دستگاه شما باید نسخه اندروید ۴/۳ یا بیشتر داشته باشد.
- گزینه **USB debugging tool** باید در موبایل شما فعال شده باشد.

نرم‌افزار systrace در آدرس `[android-sdk-directory]/platform-tools/systrace` قابل دسترسی است. فرمت دستور اجرای این نرم‌افزار بصورت زیر است:

```
$ python systrace.py [options] [categories]
```

بطور مثال:

```
$ python systrace.py -o mynewtrace.html sched freq idle am wm gfx view \
binder_driver hal dalvik camera input res
```

حتما [این لینک](#) درباره Systrace را مطالعه کنید⁴.

پس از اجرای نرم افزار، خروجی آن در یک فایل با پسوند html ذخیره می گردد.

6. سوالات

1. از وقتی که درخواست خواندن داده به سنسور داده شده تا گرفتن داده چه اتفاقاتی در سطح سیستم عامل افتاده است؟ توضیح خود را با خروجی systrace توضیح داده و توجیه کنید.
2. چه مدت زمانی طول می کشد تا مکان جدید دایره بر اساس مقدار جدیدی که از سنسور خوانده شده است، روی صفحه نمایش ظاهر شود؟ (تصویر واضح از systrace فراموش نشه ☺)
3. بنظر شما بهترین دوره تناوب برای خواندن مقادیر سنسورها و محاسبه مکان گوی چه مقدار است؟
4. بنظر شما اگر از Android NDK بجای Android SDK استفاده می شد، بازی شما چه مزایا و معایبی داشت؟
5. در مورد سنسورهای hardware-based و software-based تحقیق نمایید و هر یک را تشریح نمایید. هر کدام از سنسورهای مورد استفاده در این تمرین در کدام دسته قرار می گیرند.
6. تفاوت سنسور های Gravity و Gyroscope را تشریح نمایید. این تفاوت ها چه تاثیری بر محاسبات شما داشته است؟
7. در صورتی که بازی در حالتی شروع شود که گوشی روی سطح شیبدار قرار داشته باشد، چه اتفاقی می افتد؟ در این حالت آیا تفاوتی میان استفاده از سنسور Gyroscope و Gravity وجود دارد یا خیر؟ توضیح دهید.

⁴ اگر لازم داشتید این [لینک](#) رو هم بخونید.

7. نکات مهم

- برای پیاده‌سازی این تمرین تنها می‌توانید از زبان‌های Java و C++ استفاده نمایید.
- گزارش کار باید کامل باشد و نمره‌دهی طبق آن صورت خواهد گرفت. لذا برای نوشتن آن و پاسخ به سوالات مطرح شده وقت کافی بگذارید.
- این تمرین تحویل حضوری دارد.
- کد شما باید روی گوشی واقعی تست شده باشد.
- بازی بر روی یک دستگاه فیزیکی با اندروید 6 و بالاتر تست خواهد شد.
- بسیاری از مباحث موردنیاز برای انجام این تمرین در کلاس تی ای درباره سیستم عامل اندروید، برنامه نویسی اندروید، systrace و راه‌اندازی ابزارهایی که لازم دارید، مرور خواهد شد. خوب هست که شرکت کنید.
- حتما یک V_P_N داشته باشید!!! برای نصب Android Studio و خواندن منابع لازم‌تون میشه.
- هرگونه شباهت در کدها و گزارش‌ها به عنوان تقلب به اساتید درس گزارش خواهد شد.
- تسلط به تمام بخش‌های تمرین از راه اندازی ابزارها تا کدهای نوشته شده الزامی است و در تحویل حضوری سوال پرسیده خواهد شد.
- در صورتیکه هیچ تجربه‌ای در نوشته برنامه اندروید ندارید از همین امروز تمرین را شروع کنید تا از نظر زمانی به مشکل نخورید. از [این لینک](#) هم برای نوشتن اولین برنامه‌تون استفاده کنید و بعد تمرین رو شروع کنید.
- برای آشنایی با محیط Android Studio می‌تونید [این صفحه](#) رو ببینید.
- هر سوالی هم که داشتید رو توی فروم درس پرسید. اگر هم سوال شخصی تر داشتید می‌تونید به هر کدام از ما ۲ نفر ایمیل بزنید و در صورت نیاز حضوری با هم صحبت کنیم.

موفق باشید