

دانشگاه تهران  
دانشکده‌ی مهندسی برق و کامپیوتر



سیستم‌های نهفته‌ی بی‌درنگ

تمرین سوم

## Spinner, An Android Game

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

طراحان:

محمد حسین مطیع ([m.h.motie@gmail.com](mailto:m.h.motie@gmail.com))

روح الله ابوالحسنی ([roohi.abol@gmail.com](mailto:roohi.abol@gmail.com))

عماد جبار ([emad.jabbarnk@gmail.com](mailto:emad.jabbarnk@gmail.com))

اساتید:

دکتر مهدی کارگهی، دکتر مهدی مدرسی

نیمسال دوم ۹۸-۱۳۹۷

## ۱. مقدمه

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 بصورت کارآمد استفاده نمود؟ آیا می توان از یک smart phone در کاربرد ها صنعتی و تجاری بجای مورد های مرسوم استفاده کرد؟ چه محدودیت هایی برای این کار وجود دارد؟ در این تمرین به پاسخ سوالات فوق خواهید رسید.

در این تمرین قرار است با سیستم عامل اندروید و برخی از 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 سنسورها و ارتباط با سخت افزار لایه پایین ایجاد می کند، آشنا خواهید شد.

## ۲. شرح تمرین

در این تمرین بازی Spinner که نسخه ساده شده بازی <sup>۱</sup>Teeter Pro است را پیاده سازی خواهید کرد.

در بازی Spinner یک گوی وزن دار (برای ساده تر شدن دایره فرض کنید) وسط صفحه نمایش قرار دارد. ابتدا یک tap از طرف کاربر بر روی صفحه نمایش انجام خواهد شد تا بازی آغاز شود. گوی بر اساس زاویه تلفن همراه نسبت به افق و تغییرات زاویه و شتاب حرکت گوی حرکت متناوب انجام می دهد. برای حرکت دادن گوی روی صفحه باید از قوانین فیزیک جهت محاسبه بردار سرعت و مکان در صفحه استفاده کنید، که در بخش ۳ بصورت مشروح توضیح داده خواهد شد.

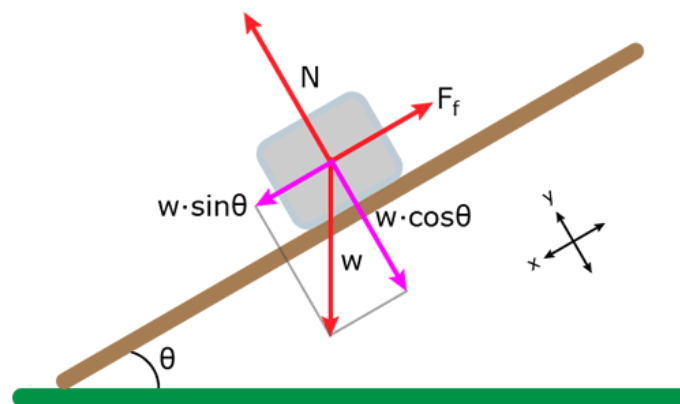
برای تشخیص جهت حرکت دایره، و سرعت دایره در صفحه از سنسور Gyroscope و Gravity<sup>۱</sup> استفاده خواهید کرد. در بخش ۴ به نحوه استفاده از این سنسورها اشاره خواهیم کرد.

<sup>۱</sup> [https://play.google.com/store/apps/details?id=pl.surix.teeterpro&hl=en\\_US](https://play.google.com/store/apps/details?id=pl.surix.teeterpro&hl=en_US)

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 گوی در صفحه نمایش حرکت کند را profile کنید و با توجه به نتایج آن، به سوالات بخش ۶ در گزارش خود پاسخ دهید.

### ۳. پیش زمینه – قوانین فیزیکی مسئله

برای حرکت دادن گوی در صفحه نیاز است تا قوانین فیزیک حاکم بر مسئله تعیین گردد. در این تمرین وزن گوی ۱۰ گرم است. همچنین گوی روی یک صفحه با ضرایب اصطکاک ایستایی  $\mu_s = 0.15$  و دینامیک  $\mu_k = 0.10$  قرار دارد. در شکل زیر، دیاگرام جسم آزاد این 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 ۱

در شکل ۱ زاویه  $\theta$  در واقع یکی از زاویه های اندازه گرفته شده توسط سنسور Gyroscope 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.

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

در زیر بعضی از قوانین فیزیک مورد نیاز آمده است:

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

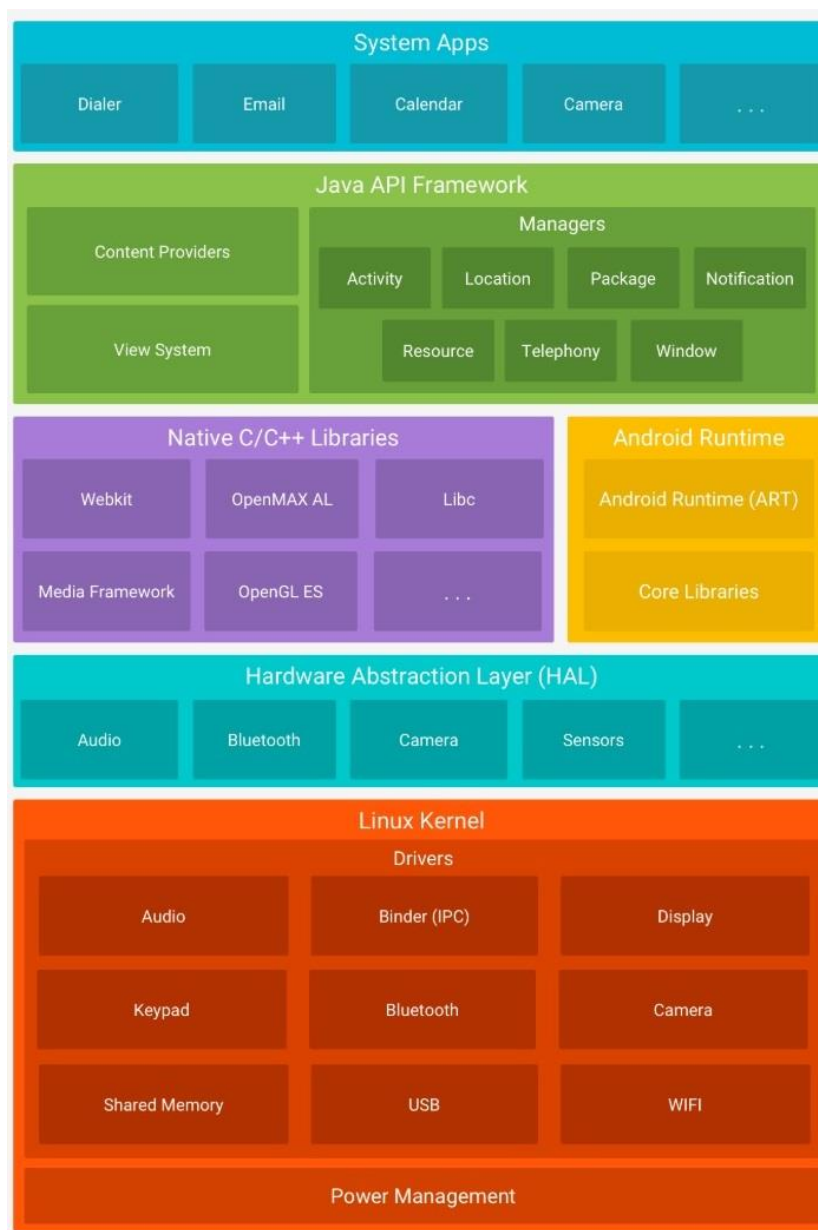
با توجه به اینکه در واقع شتاب اعمال شده به گوی ثابت نیست، بنابراین نیاز است تا بطور تقریبی حرکت گوی شبیه‌سازی شود. به این صورت که شتاب گوی در فواصل زمانی کوچک ثابت در نظر گرفته شود و محاسبات بردار سرعت و مکان بر اساس آن شتاب انجام شود. مقدار این فاصله زمانی را با سعی و خطا پیدا کنید. (دقت کنید که حرکت گوی باید بصورت پیوسته احساس شود).

#### ۴. پیاده سازی

در ابتدا قدری با معماری سیستم عامل اندروید آشنا می‌شویم تا درک بهتری نسبت به اثرات سیستم عامل بر نحوه تعامل با سنسور داشته باشیم.

سیستم عامل اندروید دارای یک معماری لایه‌ای مطابق با شکل ۱۲ است. در پایین‌ترین سطح کرنل لینوکس قرار دارد که وظیفه مدیریت منابع را برعهده دارد. سایر لایه‌ها تحت عنوان Android Open Source Project با به اختصار AOSP توسعه یافته‌اند (نسخه کرنل لینوکس مورد استفاده تفاوت‌های اندکی با کرنل vanilla دارد که در این مجال نمی‌گنجد). لایه Hardware Abstraction Layer (HAL) یک رابط استاندارد جهت تعامل با سخت‌افزارها مختلفی است که توسط شرکت‌های گوناگون ساخته می‌شود. این شرکت‌ها رابط بیان شده را پیاده‌سازی می‌کنند تا بدین ترتیب محدودیتی جهت سخت‌افزار مورد استفاده در گوشی بوجود نیاید و از مشکلاتی که بواسطه تنوع سازندگان سخت‌افزار پیش می‌آید، جلوگیری شود. در لایه بالاتر کتابخانه‌های Native که به زبان C++ نوشته شده‌اند قرار دارند که دسترسی‌های سطح پایین به امکانات کرنل را فراهم می‌کنند. در کنار این لایه ماشین مجازی Java به نام ART وجود دارد. هر برنامه یک instance از این ماشین را برای خود می‌گیرد و ریسسه‌های سطح کاربر را در این ماشین مجازی ایجاد و مدیریت می‌کند. در بالای این لایه

Java Framework قرار دارد. این لایه امکانات گوناگونی را در قالب API به توسعه‌دهندگان ارائه می‌دهد که می‌توانند با استفاده از آن از امکانات لایه‌های پایین‌تر و کرنل استفاده کنند و در عین حال برنامه‌های خود را بصورت سطح بالا پیاده‌سازی نمایند.



شکل ۲: معماری اندروید

معماری نمایش داده شده در شکل ۲ مربوط به اندروید ۵ و بعد از آن می‌باشد. البته به تناسب نسخه‌ای که در اختیار شماست ممکن است تفاوت‌های جزئی وجود داشته باشد.

برای پیاده سازی این بازی از دو سنسور استفاده خواهید کرد:

- 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 نام<sup>۲</sup> کنید. برای این کار از متد `registerListener` در کلاس `SensorManager` استفاده نمایید. در شکل ۳ 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 ۳

- 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 ۴ مشاهده نمایید:

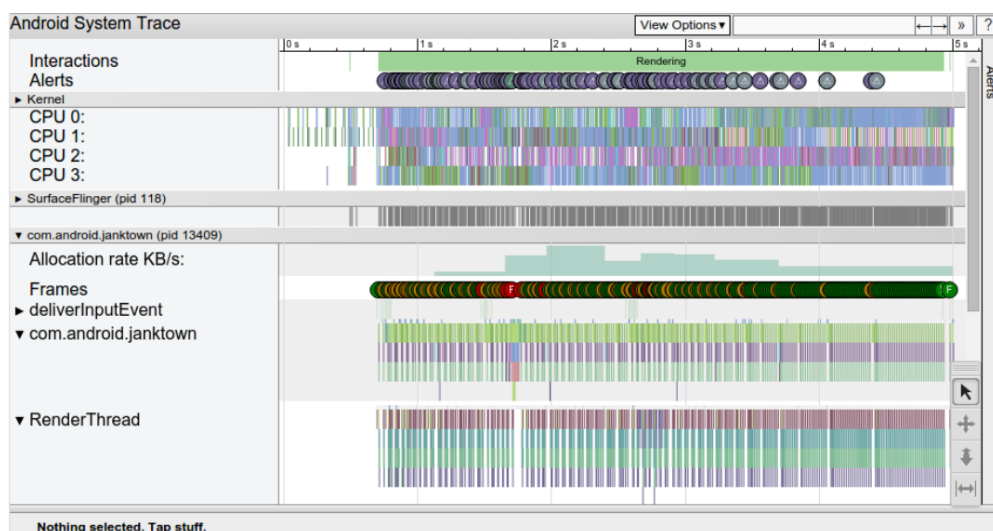
TYPE_GRAVITY	SensorEvent.values[0]	Force of gravity along the x axis.	m/s <sup>2</sup>
	SensorEvent.values[1]	Force of gravity along the y axis.	
	SensorEvent.values[2]	Force of gravity along the z axis.	

شکل ۴

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.

## ۵. ابزار profile

در یک سیستم بی درنگ تاخیر بسیار اهمیت دارد، بنابراین باید بتوان اتفاقات درون سیستم عامل را مشاهده کرد و سناریوهای مختلف را جهت اشکال زدایی از سیستم و ارائه تضمین در مورد میزان تاخیر برنامه‌های بی درنگ بررسی نمود. از آنجا که برای خواندن داده از سنسور لایه های مختلفی از سیستم عامل درگیر می‌شوند و این امر موجب ایجاد تاخیر می‌گردد در این بخش می‌خواهیم تا با کمک ابزار هایی به بررسی و مشاهده علل تاخیر، میزان آن و قسمت هایی از سیستم عامل که در این کار دخالت دارند بپردازیم و با نحوه profile کردن سیستم عامل آشنا شویم. ابزار Systrace نرم‌افزاریست جهت آنالیز کارایی دستگاه‌های اندرویدی. این ابزار متعلق به تیم‌های Chrome و Android شرکت گوگل است و در واقع بعنوان بخشی از پروژه Catapult توسعه داده شده است.<sup>۳</sup> دستور systrace به شما امکان جمع‌آوری اطلاعات زمانی پردازش‌های مختلف سیستم که در حال اجرا در سطح



شکل ۵

سیستم هستند را می‌دهد. این ابزار اطلاعات مختلف کرنل اندروید مانند CPU scheduler و app threadها را جمع‌آوری کرده و در یک فایل html مشابه شکل ۵ بعنوان خروجی نشان می‌دهد.

همان طور که در تصویر فوق مشاهده می‌کنید، هر ریسسه<sup>۴</sup> در این برنامه یک نوار مختص به خود دارد که وضعیت آن ریسسه در زمان را نمایش می‌دهد. (در این مثال ۵ ثانیه است که در نوار بالای تصویر مشخص شده است) 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 2.7 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.
- حتما ADB بر روی سیستم شما نصب 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
```

در مثال فوق پس از -o نام فایل خروجی آمده است. محل ذخیره فایل در همان مسیر فایل systrace.py می‌باشد. در ادامه<sup>۵</sup> categoryهای مختلفی که قرار است trace شوند آمده است. برای اینکه لیست category

<sup>۴</sup> Thread

<sup>۵</sup> منظور از category سرویس‌های مختلفی است که سیستم عامل اندروید ارائه می‌دهد.



هایی را که دستگاه شما پشتیبانی می‌کند ملاحظه کنید در زمانی که گوشی به سیستم شما متصل است دستور زیر را در ترمینال وارد نمایید:

```
python systrace.py --list-categories
```

برای تعیین مدت زمان trace از -t که پس آن مدت زمان trace کردن با واحد ثانیه می‌آید، می‌توانید استفاده نمایید. از آنجایی که systrace برای نگهداری رویدادها از یک بافر استفاده می‌کند شما می‌توانید حجم این بافر را با -b که پس از آن سایز بافر با واحد کیلوبایت می‌آید محدود کنید. در صورتی که مشابه مثال سایز را مشخص نکنید ابزار تمام رویدادها را با buffer می‌کند که می‌تواند حجم زیادی پیدا کند.

برای آشنایی بیشتر با systrace و همچنین مشاهده option های موجود می‌توانید به این [لینک](#) مراجعه نمایید.<sup>۶</sup>

## ۶. سوالات

۱. از وقتی که درخواست خواندن داده به سنسور Gyroscope و Gravity داده شده تا گرفتن داده چه اتفاقاتی در سطح سیستم عامل افتاده است؟ تو ضیح خود را با خروجی systrace تو ضیح داده و توجیه کنید. (برای هم سنسور جداگانه این کار را انجام دهید).
۲. چه مدت زمانی طول می‌کشد 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 😊)
۳. بنظر شما اگر از Android NDK بجای Android SDK استفاده می‌شد، بازی شما چه مزایا و معایبی داشت؟
۴. 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 می‌گیرند.
۵. تفاوت سنسورهای Gravity و Gyroscope را تشریح نمایید. این تفاوت‌ها چه تاثیری بر محاسبات شما داشته است؟

<sup>۶</sup> اگر لازم داشتید این [لینک](#) رو هم بخونید.

۶. 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.

## ۷. نکات مهم

- برای پیاده سازی این تمرین تنها می توانید از زبان های Java و C++ استفاده نمایید.
- گزارش کار باید کامل باشد و تصحیح از روی آن صورت خواهد گرفت. لذا برای نوشتن آن و پاسخ به سوالات مطرح شده دقت کافی را مبذول دارید.
- این تمرین تحویل حضوری دارد
- در زمان تحویل حضوری طراحی مفهومی خود را همراه داشته باشید. (چه روی کاغذ و چه روی کامپیوتر.)
- کد شما باید روی گوشی واقعی تست شده باشد.
- بازی بر روی یک دستگاه فیزیکی با اندروید ۶ و بالاتر تست خواهد شد.
- کلی آموزش توی کلاس تی ای درباره سیستم عامل اندروید، برنامه نویسی اندروید، systrace و راه اندازی ابزارهایی که لازم دارید، داریم.
- حتما یک V\_P\_N (خوب ☺) داشته باشید!!! برای نصب Android Studio و خواندن منابع لازمتون میشه.
- هرگونه شباهت در کدها و گزارش ها به عنوان تقلب به اساتید درس گزارش خواهد شد.
- تسلط به تمام بخش های تمرین از راه اندازی ابزارها تا کدهای نوشته شده الزامی است و در تحویل حضوری سوال پرسیده خواهد شد.
- در صورتیکه هیچ تجربه ای 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 برنامه تون استفاده کنید و بعد تمرین رو شروع کنید.
- برای آشنایی با محیط Android Studio می تونید [این صفحه](#) رو ببینید.

- هر سوالی هم که داشتید رو توی فروم درس پرسید. اگر هم سوال شخصی تر داشتید می تونید به هر کدوم از ما ۳ نفر ایمیل بزنید و در صورت نیاز حضوری با هم صحبت کنیم.

**موفق باشید**