به نام خدا

دانشگاه تهران

دانشکده‌ی مهندسی برق و کامپیوتر

سیستم‌های نهفته‌ی بی‌درنگ

تمرین موس هوایی

(Air Mouse)

با هدف آشنایی با سیستم عامل اندروید و استفاده از سنسورهای تلفن همراه

اساتید:

دکتر مهدی کارگهی، دکتر محسن شکری ساز

طراحان:

ارسلان طلائی، آرین فیروزی

نیمسال دوم ۱۴۰۴-۱۴۰۵

۱. مقدمه

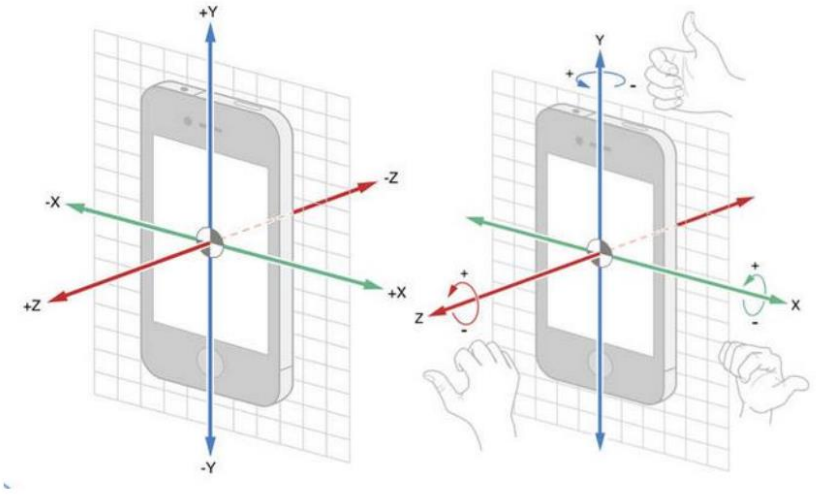
امروزه تلفن‌های همراه هوشمند پیشرفت چشمگیری داشته‌اند و به پردازنده‌های قدرتمند، مجموعه‌ای متنوع از سنسورها و قابلیت پشتیبانی از شبکه‌های ارتباطی مختلف مجهز شده‌اند. علاوه بر این، همه این قابلیت‌ها در قالب دستگاه‌هایی کوچک، سبک و همیشه در دسترس کاربران قرار گرفته‌اند.

در این میان پرسش مهمی مطرح می‌شود: چگونه می‌توان از این امکانات که همواره همراه کاربر هستند بهره‌برداری کرد؟ آیا امکان استفاده از تلفن همراه در کاربردهای نو و تجاری به‌عنوان جایگزینی برای سامانه‌های نهفته‌ی سنتی وجود دارد؟ همچنین در صورت استفاده از تلفن همراه برای چنین کاربردهایی، چه محدودیت‌ها و چالش‌هایی مطرح خواهد بود؟

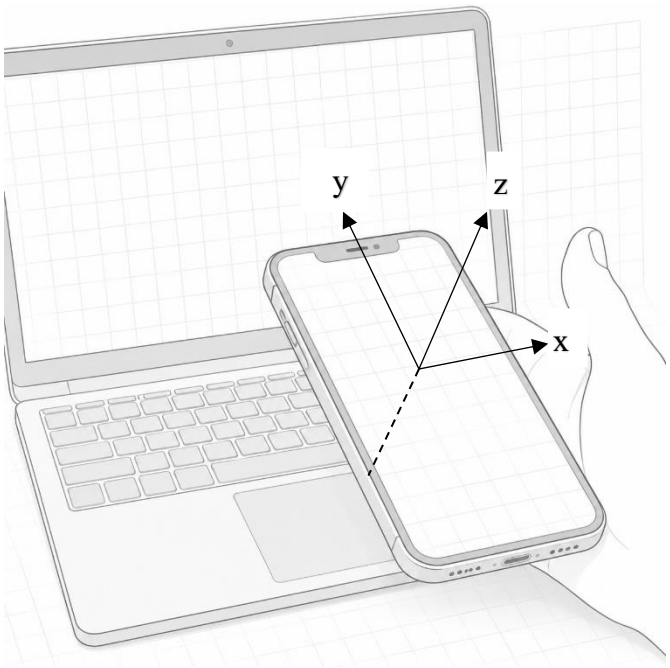
در این تمرین، هدف آشنایی با سیستم‌عامل اندروید و قابلیت‌هایی است که این سیستم‌عامل برای تعامل با سنسورهای تلفن همراه در اختیار برنامه‌نویسان قرار می‌دهد. همچنین به بررسی برخی محدودیت‌های سیستم‌عامل در دسترسی به سنسورها و برقراری ارتباط با لایه‌های پایین‌تر سخت‌افزار خواهیم پرداخت.

۲. شرح تمرین

هدف این تمرین، آن است که با استفاده از سنسورهای یک تلفن هوشمند مبتنی بر سیستم‌عامل اندروید، یک Air Mouse طراحی و پیاده‌سازی شود. در این سامانه، حرکت و جهت‌گیری تلفن همراه در فضا به‌عنوان ورودی در نظر گرفته می‌شود و از آن برای کنترل اشاره‌گر یا اجرای فرمان‌های حرکتی استفاده خواهد شد. (برای آشنایی با سنسور ها، **پیوست ۱ را بخوانید.**)



شکل ۱ - جهت بردارهای ژیروسکوپ و شتابسنج بر روی میز



شکل ۲ - نمایش وضعیت قرارگیری تلفن همراه در دست کاربر و محورهای حرکتی آن برای کنترل ماوس هوایی

در این تمرین فرض می‌شود کاربر تلفن هوشمند را به‌صورت عمود بر صفحه‌نمایش لپ‌تاپ و موازی سطح کیبورد (مانند تصویر) در دست می‌گیرد؛ به‌گونه‌ای که حرکت و چرخش گوشی در فضا بتواند به فرمان‌های کنترلی ماوس روی صفحه تبدیل شود.

در این سیستم، حرکت گوشی حول دو محور برای کنترل جابه‌جایی نشانگر ماوس استفاده می‌شود. به‌طور مشخص، تغییرات حرکتی در راستای محورهای Z و X به‌ترتیب برای جابه‌جایی افقی و عمودی نشانگر ماوس روی صفحه در نظر گرفته می‌شوند. بنابراین، با تغییر وضعیت گوشی در راستای محور Z، نشانگر ماوس در جهت افقی حرکت می‌کند و با تغییر وضعیت گوشی در راستای محور X، نشانگر ماوس در جهت عمودی جابه‌جا می‌شود.

همچنین از حرکت یا تغییر وضعیت در راستای محور Y برای تشخیص عمل کلیک استفاده می‌شود. به این ترتیب، در صورتی که کاربر چرخش سریع به سمت چپ حول محور Y انجام دهد ، سیستم آن را به‌عنوان فرمان کلیک تشخیص داده و سیگنال ماوس، یعنی کلیک چپ، را ارسال می‌کند. در نتیجه، کاربر می‌تواند بدون نیاز به ماوس فیزیکی و تنها با حرکت دادن گوشی در فضا، نشانگر ماوس را کنترل کرده و عملیات کلیک را انجام دهد.

علاوه بر جابه‌جایی نشانگر و تشخیص کلیک، قابلیت اسکرول کردن صفحه نیز باید پیاده‌سازی شود. برای این منظور، جابه‌جایی سریع گوشی در راستای محور مربوط به کلیک، یعنی محور Y، به‌عنوان فرمان اسکرول در نظر گرفته می‌شود. در این حالت، اسکرول به‌صورت دو جهته انجام می‌شود؛ به این معنا که حرکت سریع گوشی در یک جهت از محور Y فرمان اسکرول به پایین و حرکت سریع در جهت مخالف همان محور فرمان اسکرول به بالا را ایجاد می‌کند. بنابراین، تفاوت اسکرول بالا و پایین بر اساس علامت و جهت تغییرات محور Y تعیین می‌شود. برای مثال، عبور تغییرات محور Y از یک آستانه مشخص در جهت مثبت می‌تواند به‌عنوان اسکرول به بالا و عبور آن در جهت منفی به‌عنوان اسکرول به پایین تفسیر شود. البته جهت دقیق مثبت و منفی محور، بسته به نحوه قرارگیری گوشی و تعریف سنسورها، باید در مرحله آزمایش تنظیم شود.

نکته مهم این است که حرکت اسکرول باید در راستای محور Y انجام شود و تشخیص آن بر اساس داده‌های همان محور باشد؛ بنابراین صرفاً ذخیره‌سازی یک الگوی حرکتی کلی یا ترکیبی، بدون توجه به راستای حرکت روی محور Y، کافی نیست و ملاک ارزیابی قرار نمی‌گیرد. همچنین برای جلوگیری از اجرای ناخواسته اسکرول، باید یک آستانه سرعت یا مقدار تغییر مشخص تعریف شود تا لرزش‌های کوچک دست و نویز سنسورها نادیده گرفته شوند. علاوه بر این، پس از تشخیص هر اسکرول می‌توان یک فاصله زمانی کوتاه یا حالت انتظار در نظر گرفت تا حرکت برگشت دست به وضعیت اولیه، به‌اشتباه به‌عنوان اسکرول در جهت مخالف تشخیص داده نشود.

در نهایت، عملکرد سیستم باید به این صورت باشد که کاربر با حرکت گوشی در راستای محورهای Z و X بتواند نشانگر ماوس را به‌ترتیب در جهت افقی و عمودی جابه‌جا کند، با چرخش به سمت چپ در راستای محور Y عملیات کلیک چپ را انجام دهد و با جابه‌جایی سریع در راستای همان محور Y، بسته به جهت حرکت، صفحه را به سمت بالا یا پایین اسکرول کند.

- **دقت در حرکت**

یکی از پارامترهای مهم در ارزیابی پروژه، دقت و پایداری عملکرد ماوس در هنگام حرکت است. همچنین، به‌منظور ارزیابی یکسان برنامه‌ها و بی‌اثر کردن برتری احتمالی یک سنسور نسبت به سنسور دیگر، تمامی برنامه‌ها روی یک تلفن همراه واحد ارزیابی خواهند شد. بنابراین، لازم است راهکارهای مناسب برای مقابله با انواع نایقینی‌ها در تفسیر مقادیر خوانده‌شده از سنسورها به کار گرفته شود.

از جمله این نایقینی‌ها می‌توان به نویز، Drift و Bias اشاره کرد. برای مثال، یکی از مشکلات رایج این است که با وجود قرار داشتن گوشی در حالت ایستای اولیه، نشانگر ماوس همچنان حرکت کند.

به‌منظور کاهش یا رفع مشکلات گفته‌شده، می‌توانید راهکارهای زیر را بررسی و پیاده‌سازی کنید:

- کالیبره کردن سنسورها پیش از شروع به کار
- استفاده از فیلترهای بالاگذر و پایین‌گذر روی مقادیر خوانده‌شده از سنسورها

- **ترکیب حسگرها**

می‌توانید از فیلترهای متداول مانند Mahony، Madgwick، Kalman و فیلتر مکمل استفاده کنید. این فیلترها نمونه‌هایی از الگوریتم‌های AHRS یا Attitude and Heading Reference System هستند. برای آشنایی بیشتر با این روش‌ها، مطالعه **پیوست‌های ۲، ۳، ۴ و ۵** توصیه می‌شود.

دقت داشته باشید که استفاده از کتابخانه‌ها و راه‌حل‌های آماده مجاز نیست. بنابراین، در صورت انتخاب هر یک از این روش‌ها، باید پیاده‌سازی آن را متناسب با صورت مسئله و به‌صورت مستقل ارائه کنید.سنسورهای قابل استفاده در این بخش عبارت‌اند از: جاذبه‌سنج، شتاب‌سنج، سنسور مغناطیسی و دوران‌نما (ژیروسکوپ).

۳. پیاده‌سازی

پیاده‌سازی این سؤال شامل دو بخش است: بخش نخست در سمت تلفن همراه قرار دارد و وظیفه دریافت و پردازش داده‌های سنسورها را بر عهده دارد؛ بخش دوم نیز در سمت لپ‌تاپ یا رایانه شخصی اجرا می‌شود و برای نمایش و کنترل مکان‌نمای ماوس مورد استفاده قرار می‌گیرد.

- **ابزارهای مورد نیاز**

برای پیاده‌سازی این تمرین، ابتدا لازم است نرم‌افزار **Android Studio** را روی سیستم خود نصب کنید. راهنمای نصب این نرم‌افزار در اختیار شما قرار گرفته است. با توجه به محدودیت دسترسی به اینترنت بین‌الملل و دشواری دریافت برخی وابستگی‌ها، لازم است مراحل نصب را دقیقاً مطابق دستورالعمل ارائه‌شده انجام دهید. این دستورالعمل در **پیوست شماره ۶** قرار دارد.

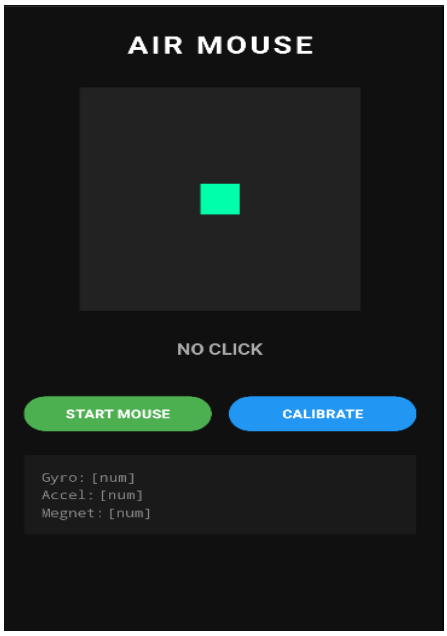
روش نصب ارائه‌شده از قبل آزمایش شده و به‌درستی کار می‌کند. در صورتی که هنگام نصب یا اجرای نرم‌افزار با مشکلی مواجه شدید، می‌توانید با دستیاران آموزشی درس در ارتباط باشید. پس از نصب موفق **Android Studio**، لازم است با استفاده از ابزارها و کتابخانه‌های مربوط به سنسورها در این محیط، نحوه دسترسی به سنسورهای دستگاه و استفاده از آن‌ها را مطالعه کرده و در پروژه خود به‌کار ببرید. با توجه به محدودیت‌های دسترسی، فایل‌های آموزشی مربوط به نحوه کار با سنسورها نیز تهیه شده‌اند و در پیوست‌های تمرین قابل مشاهده هستند. (**پیوست شماره ۷و ۸ را مطالعه نمایید.**)

همچنین یک ویدئوی آموزشی به این تمرین ضمیمه شده است؛ این ویدئو در **پیوست شماره ۱۶** قرار دارد و در آن عملکرد هر یک از سنسورهای موردنیاز این سؤال، نحوه ترکیب داده‌های آن‌ها، و روش اعمال فیلتر بر خروجی سنسورها برای دستیابی به خروجی دقیق‌تر و مطلوب‌تر توضیح داده شده است. توصیه می‌شود پیش از شروع پیاده‌سازی، حتماً این ویدئو را مشاهده کنید.

توجه داشته باشید که تمرین روی دستگاه اندرویدی اجرا خواهد شد؛ بنابراین نسخه اندروید پروژه را به‌گونه‌ای تنظیم کنید که خروجی کد شما روی **اندروید ۱۰ و نسخه‌های بالاتر** قابل اجرا باشد.

- **سمت اندروید**

برنامه ای که برای این تمرین مینویسید بر روی گوشی از ۴ بخش تشکیل شده است.



شکل ۳ – نمونه رابط کاربری (نمونه ساده که شامل تمام جزئیات گفته شده **نمی‌باشد.**)

۳-۱ رابط کاربری

برای استفاده از اپلیکیشن، لازم است رابط کاربری مناسب با نیاز های گفته شده طراحی شود. رابط کاربری شما باید شامل یک دکمه برای **کالیبره کردن** و دکمه‌ای دیگر برای **شروع ارسال اطلاعات** باشد. همچنین لازم است جهت حرکت ماوس در رابط کاربری نمایش داده شود؛ برای مثال، با چرخاندن گوشی، مربع سبزرنگ موجود در شکل نمونه باید در همان جهت چرخش گوشی حرکت کند.

علاوه بر این، عمل **کلیک** و **اسکرول** نیز باید به‌نحوی در رابط کاربری نمایش داده شود. برای نمونه، استفاده از یک **Label** یا چشمک‌زدن مربع می‌تواند گزینه مناسبی برای نمایش وقوع کلیک و انجام عمل اسکرول باشد.

همچنین، برای آسان‌تر شدن فرایند دیباگ و تست، مقادیر خوانده‌شده از سنسورهای مختلف باید در رابط کاربری نمایش داده شوند.

توجه داشته باشید که رابط کاربری باید به‌گونه‌ای طراحی شود که با تغییر اندازه نمایشگر، قالب آن به‌هم نریزد و همچنان قابل استفاده باشد.

برای انجام کالیبراسیون نیز به صفحه‌ای جداگانه نیاز است. در این صفحه باید به کاربر توضیح داده شود که در هر مرحله چه کاری انجام دهد تا سنسورها به‌درستی کالیبره شوند. جزئیات مربوط به مراحل کالیبراسیون در بخش بعدی توضیح داده شده است.

نکته: با توجه به پویا بودن IP لپ‌تاپ، توصیه می‌شود یک فیلد برای وارد کردن IP لپ‌تاپ در رابط کاربری اندروید در نظر گرفته شود.

۳-۲ سنسورها و فیلترهای مربوطه

در این مرحله، شما باید اطلاعات را از سنسورهای **خام** بخوانید و بر اساس آن، جهت حرکت گوشی را تشخیص دهید. استفاده از سنسورهای کالیبره‌شده‌ای که اندروید به‌صورت پیش‌فرض ارائه می‌دهد، مجاز نیست. برای آشنایی بیشتر، به پیوست‌های ۷ و ۸ مراجعه کنید.

همان‌طور که در شرح تمرین توضیح داده شد، شما باید با استفاده از فیلترهای مختلف این سنسورها را کالیبره کرده و دریفت آن‌ها را برطرف کنید.

زاویه سنج[۱]: برای رفع **بایاس** این سنسور، ابتدا باید آن را در حالت ثابت قرار دهید و چندین نمونه از آن دریافت کنید. سپس میانگین مقدار گزارش‌شده را محاسبه کنید؛ مقداری که در حالت ایده‌آل باید برابر صفر باشد. با کم‌کردن این مقدار از مقادیر اندازه‌گیری‌شده، می‌توانید با دقت نسبتاً خوبی به مقدار واقعی برسید

شتاب سنج[۲]: برای کالیبره‌کردن این سنسور، باید گوشی را در شش وضعیت مختلف قرار دهید و خطای مقادیر گزارش‌شده را محاسبه کنید. در هر یک از این شش حالت، یکی از سه مقدار خروجی سنسور باید با مقدار ثابت و مثبت یا منفی شتاب گرانش، یعنی ۹.۸۱، برابر باشد.

---

[۱] Gyroscope

[۲] Accelerometer

در نهایت، بر اساس این مقادیر، باید خطای offset (یعنی مقدار غیرصفر در حالتی که مقدار ایده‌آل باید صفر باشد) و نیز ضریب مقیاس (یعنی مقدار ناصفرِ نابرابر با ثابت گرانش) را به‌دست آورید. هر یک از مقادیر گزارش‌شده توسط سنسور، با استفاده از یک معادله خطی درجه اول به مقدار واقعی تبدیل می‌شوند.

مغناطیس سنج[۳]: برای کالیبره‌کردن این سنسور، کافی است گوشی را به شکل عدد ۸ انگلیسی در فضا حرکت دهید. در این حین، باید حداقل و حداکثر مقادیر گزارش‌شده برای هر یک از سه محور را ذخیره کنید.با استفاده از این داده‌ها، offset هر محور برابر با میانگین مقدار حداقل و حداکثر، و scale هر محور برابر با نصف اختلاف مقدار حداکثر و حداقل خواهد بود.. عدد تصحیح شده از فرمول (Raw - Offset) / Scale به دست می آید.

این روش برخلاف شتاب‌سنج نیاز به جهت‌گیری دقیق ندارد وفقط کافیست گوشی را در تمام زوایا بچرخانید تا نمونه‌های کافی از کل کره برداشت شود.

ترکیب: برای برای رفع دریفت سنسورها، می‌توانید از فیلتر Madgwick AHRS استفاده کنید که نسبت به الگوریتم‌های قبلی پیچیده‌تر است. نحوه‌ی پیاده‌سازی این الگوریتم در اختیار شما قرار داده شده است. همچنین، در صورت تمایل، می‌توانید از فیلترهای دیگری که مناسب می‌دانید استفاده کنید و الزامی به استفاده از فیلتر Madgwick وجود ندارد.

۳-۳ حرکت گوشی

چرخش سه‌بعدی گوشی باید به‌گونه‌ای پردازش شود که تغییر وضعیت آن در دو راستا باعث جابه‌جایی نشانگر ماوس روی صفحه شود و حرکت یا چرخش سریع در راستای محور سوم، برای تشخیص فرمان‌های مجزا مانند کلیک و اسکرول مورد استفاده قرار گیرد. به‌طور مشخص، تغییر وضعیت گوشی در دو محور اصلی باید به جابه‌جایی افقی و عمودی نشانگر تبدیل شود، در حالی که حرکت سریع در محور دیگر می‌تواند برای ارسال سیگنال کلیک یا فرمان اسکرول به کار رود.

برای پیاده‌سازی اسکرول، حرکت سریع گوشی در راستای محور مربوط به اسکرول باید تشخیص داده شود؛ به‌طوری‌که حرکت در جهت مثبت این محور به‌عنوان اسکرول به بالا و حرکت در جهت منفی آن به‌عنوان اسکرول به پایین در نظر گرفته شود. تشخیص اسکرول باید تنها زمانی انجام شود که حرکت به‌طور مشخص در راستای همان محور رخ داده باشد تا از تداخل آن با حرکت نشانگر یا فرمان کلیک جلوگیری شود. همچنین، برای تشخیص کلیک نیز می‌توان از حرکت یا چرخش سریع حول محور تعیین‌شده استفاده کرد.

توجه داشته باشید که در این بخش بهتر است حساسیت گوشی به‌صورت یک پارامتر قابل تنظیم در نظر گرفته شود تا در صورت نیاز بتوان آن را متناسب با شرایط تغییر داد. همچنین، مقادیر ارسالی نباید به‌صورت مقادیر مطلق باشند؛ بلکه باید اختلاف مقدار فعلی و مقدار قبلی نشانگر ارسال شود. برای افزایش دقت و جلوگیری از تشخیص‌های اشتباه، به‌ویژه در مورد کلیک و اسکرول، می‌توانید از آستانه‌ی سرعت یا مقدار تغییر و نیز سازوکارهایی به انتخاب خودتان استفاده کنید تا حرکت‌های برگشتی دست به‌اشتباه به‌عنوان فرمان مستقل تشخیص داده نشوند

---

[۳] Magnetometer

۳-٤ ارسال داده ها به لپ‌تاپ

در این بخش، برنامه‌ی اندرویدی باید داده‌های پردازش‌شده‌ی سنسورها را از طریق شبکه به لپ‌تاپ ارسال کند. برای این منظور، نکات زیر باید رعایت شود:

- یک پورت شبکه مشخص، مانند ۵۰۰۰ یا ۸۰۸۰، برای برقراری ارتباط در نظر گرفته شود.
- تبادل داده‌ها از طریق Socket Programming و با استفاده از پروتکل UDP انجام شود.
- اطلاعات ارسالی می‌تواند شامل مقادیر مربوط به محورهای حرکتی و وضعیت کلیک باشد و به‌صورت رشته‌ای، برای مثال در قالب JSON، ارسال شود.به‌عنوان نمونه، یکی از روش‌های ارسال می‌تواند شامل پارامترهای DeltaX، DeltaY و Click ، Scroll باشد.
- پس از ارسال بسته‌هایی که شامل سیگنال کلیک یا اسکرول هستند، لازم است ACK از سمت گیرنده دریافت شود.

از آن‌جا که نرخ ارسال داده در گوشی و نرخ دریافت در لپ‌تاپ ممکن است با یکدیگر متفاوت باشد، باید این اختلاف در طراحی سیستم در نظر گرفته شود. بسته‌هایی که صرفاً شامل اطلاعات حرکتی هستند، در صورت نیاز می‌توانند **حذف** (drop) شوند؛ اما در صورتی که بسته شامل **سیگنال کلیک** یا **اسکرول** باشد، این داده باید تا زمان اطمینان از دریافت آن توسط سرور محفوظ بماند و در صورت لزوم، مجدداً ارسال شود.

همچنین توجه داشته باشید که برای برقراری ارتباط بین گوشی و رایانه، هر دو دستگاه باید به یک **شبکه مشترک**، مانند یک مودم یا شبکه‌ی محلی، متصل باشند. علاوه بر این، لازم است مجوز دسترسی به اینترنت و نیز اجازه‌ی استفاده از ارتباط ناامن (cleartext) در برنامه فعال شود. این کار از طریق دستورات زیر در فایل Manifest قابل انجام است:

```
<uses-permission android:name="android.permission.INTERNET" />
<base-config cleartextTrafficPermitted="true" />
```

- **سمت لپ‌تاپ یا رایانه شخصی:**

در سمت لپ‌تاپ لازم است برنامه‌ای نوشته شود که:

۱. بر روی پورت شبکه‌ای که از پیش مشخص شده است، در حالت انتظار برای دریافت داده قرار گیرد.
۲. داده‌های ارسالی از سمت تلفن همراه را دریافت کند.
۳. داده‌های دریافتی را پردازش و تفسیر کند.
۴. بر اساس مقادیر دریافت‌شده، مکان‌نمای ماوس را روی صفحه جابه‌جا کند و در صورت نیاز، عملیات کلیک یا اسکرول را نیز انجام دهد.
۵. در صورت لزوم، بسته‌ای برای تأیید دریافت صحیح سیگنال کلیک به فرستنده ارسال کند..

برای پیاده‌سازی این بخش، استفاده از زبان Python پیشنهاد می‌شود؛ زیرا پیاده‌سازی سوکت در این زبان ساده‌تر است و کتابخانه‌هایی برای کنترل ماوس، مانند PyAutoGUI، در دسترس هستند. برای نصب این کتابخانه می‌توانید از دستور زیر استفاده کنید :

```
pip install pyautogui -i https://pypi.devneeds.ir/simple/
```

خروجی: برای تحویل این تمرین، کد سورس برنامه و یک خروجی apk تحویل خواهید داد که در آن موس هوایی با توضیحات بالا ساخته شده باشد. همچمین یک ویدئو کوتاه از صحت کارکرد برنامه خود ارسال کنید. در این ویدئو صفحه لپتاپ و تلفن همراه به طور همزمان و واضح مشخص باشد. (در ویدئو صحت کارکرد کلیک، اسکرول و جابجایی مکان نما قابل مشاهده باشد.)

توجه: حرکت موس باید به نرمی و توسط چرخش دست کنترل شود و پرش نداشته باشد. و در طول زمان تجمیع خطا نباید وجود داشته باشد. حرکت های غیر ارادی و کوچک دست در هنگام ساکن بودن بر روی نمایشگر لپتاپ نباید مشاهده شود. اعمال کلیک و اسکرول نیز با دقت خوبی انجام شوند.

۴. ابزار Profile and Trace

ابزاری که شرکت گوگل برای انجام عمل Profiling و Tracing در اختیار برنامه نویسان قرار داده است، Perfetto نام دارد.(**پیوست شماره ۹** تمام داکیومنتیشن سایت Perfetto است.)این ابزار به ما کمک می‌کند تا بتوانیم از برنامه‌ای که نوشتیم، گزارش تهیه کنیم. این ابزار با نمونه برداری از سیستم و نشان دادن این داده‌ها به صورت گرافیکی به ما کمک می‌کند که عملکرد برنامه را تجزیه و تحلیل کنیم. برای مثال، در صورتی که بخواهیم تعداد Context Switch را در زمان‌بند بدانیم، باید به پارامترهایی که زمان‌بند از آن‌ها نگهداری می‌کند دسترسی پیدا کنیم. ابزارهای بر پایه trace به این پارامترها دسترسی پیدا می‌کنند و اطلاعات مورد نیاز را برای ما فراهم می نمایند.

شکل ۴- تصویری از محیط ابزار Perfetto

یرای آشنایی بیشتر با مفاهیم مربوط به این ابزار و نحوه استفاده از آن، می‌توانید به **پیوست‌های ۱۰، ۱۱-۱، ۱۱-۲ و ۱۱-۳** مراجعه کنید.

به‌عنوان نمونه، برای شروع کار، ابتدا گزینه‌ی USB Debugging را در دستگاه اندرویدی خود فعال کنید. سپس، در سیستم‌عامل ویندوز، برای دریافت یک Trace از دستگاه، فایل پیوست‌شده با نام record_android_trace را در مسیر مشخص‌شده کپی کنید. این فایل در پیوست شماره ۱۲ قرار داده شده است.

```
C:\Users{Username}\AppData\Local\Android\Sdk\platform-tools
```

پس از کپی کردن فایل، یک cmd در همین مسیر باز کرده و دستور زیر را اجرا کنید

```
python record_android_trace -o trace_file.perfetto-trace -t 10s sched freq idle wm gfx view
```

با اجرای این دستور، یک Trace از سیستم اندروید به مدت ۱۰ ثانیه ثبت می‌شود و فایل خروجی با نام trace_file.perfetto-trace در همان مسیر platform-tools ذخیره خواهد شد. این فایل برای تحلیل عملکرد سیستم در ابزار Perfetto UI قابل استفاده است.با توجه به در دسترس نبودن مستقیم ابزار Perfetto UI، برای مشاهده و تحلیل فایل Trace باید از دستورالعمل ضمیمه‌شده استفاده کنید. در این دستورالعمل، اجرای ابزار Perfetto UI از طریق Docker Image توضیح داده شده است.(**به پیوست شماره ۱۳ مراجعه کنید)**

برای تحویل این تمرین نیاز دارید که تنظیمات این ابزار را در فایلی به نام config.pbtx بنویسید.(**به پیوست شماره ۱۴ مراجعه کنید)** سپس میتوانید بعد از انجام تنظیمات عملیات trace مورد نظر خود را انجام دهید و به سوالات خواسته شده پاسخ دهید. برای نوشتن این فایل میتوانید به فایل پیوست مربوطه مراجعه کنید. ممکن است نیاز باشد برای استفاده از این ابزارevent های خود را در سطح کد تعریف کنید که این کار میتواند بسته به زبان مورد استفاده توسط کتابخانه traceدر اندروید انجام شود.

هر تحلیل و نتیجه گیری در مورد سوالات perfetto باید با استدلال صحیح صورت گیرد. مثلابرای اندازه‌گیری استفاده از ابزار صورت چشمی یا مشخص کردن بازه توسط ماوس ملاک نیست. برای این **کار حتما از کتابخانه python ابزار perfettoو در صورت صلاح دید از کتابخانه های دیگر برای تحلیل دیتای خروجی perfetto استفاده کنید.**

نکته : در هنگام استفاده از perfetto بدون اینترنت، نیاز است به **فایل های پیوست شماره ۱-۱۵ و ۲-۱۵** با توجه به سیستم عامل خود مراجعه کنید و فایل مورد نظر از استخراج کرده و دستور زیر را در cmd در همان مسیر اجرا نمایید.

```
./trace_processor_shell.exe -httpd
```

و در کد پایتون خود، از سرور اجرا شده بدین نحو استفاده کنید.

```
// download with command:  pip install  perfetto -i https://pypi.devneed.ir/simple
from perfetto.trace_processor import TraceProcessor
```

```
tp = TraceProcessor(
    trace= “trace_file”,
    addr=http://127.0.0.1:9001)
```

توضیحات تکمیلی در پیوست شماره ۹ قابل یافتن است.

سوالات

۱) از وقتی که درخواست خواندن داده به یک سنسور داده شده تا گرفتن داده چه اتفاقاتی در سطح سیستم عامل افتاده است؟ توضیح خود را با خروجی Perfetto توضیح داده و توجیه کنید. (برای این سوال میتوانید از perfetto uiاستفاده کنید)

۲) کارکرد سنسورهای استفاده شده را شرح دهید. توضیح دهید که سنسورهای خام چرا خطا دارند و چگونه ترکیب این سنسورها به کاهش خطا کمک کرده است.

۳) زمان بین خواندن دو داده متوالی از سنسور در Perfetto را با دوره نمونه‌برداری که در کد خود پیکربندی نموده‌اید، مقایسه کنید.

۴) آیا هنگام استفاده از فراخوانی‌های سیستمی(System Calls) ، میان فعالیت‌های حساس به زمان (مانند به‌روزرسانی سنسورها) و پردازش‌های سنگین (مانند رندر گرافیکی)، تعارضی (یک thread تا زمانی که thread دیگر کار خود را تمام کند باید منتظر بماند)رخ می‌دهد؟ پاسخ خود را توجیه کنید.

۵) چه تفاوتی بین تعریف سنسور به صورت wake-up و non-wake-up وجود دارد؟ مزایا و معایب هر کدام؟

۶) مدت زمانی را که تابع فیلتر در سطح CPU اجرا میشود را به صورت میانگین به دست آورید.

۷) بیشترین مصرف پردازشی مربوط به کدام سنسور بوده است؟

۸) نرخ نمونه‌برداری سنسورها چه تأثیری بر سایر پردازش‌های سیستمی دارد؟

۹) از آماده شدن اطلاعات سنسور تا ارسال به سرور/رایانه و مشاهده حرکت مکان نما متوسط چقدر زمان مصرف می‌شود؟

۱۰) threadهای اصلی، threadهای سنسور، threadهای ارتباطی، پردازشی و UI چگونه تفکیک می‌شوند؟ بررسی شود که هر سنسور چه نوعevent هایی تولید می‌کند، اینevent ها روی کدامthread ها اجرا می‌شوند و چه تأثیری برMain Thread دارند؟

۱۱) تفاوت حرکت آرام و ناگهانی تلفن همراه روی پردازش های صورت گرفته چگونه است ؟ آیا تاخیر حرکت مکان نما ماوس در دو حالت تفاوتی دارد؟

نکات تحویل

- **مهلت ارسال پاسخ: ۱۴۰۴/۳/۲۲ ساعت ۲۳:۵۹**
- **گروه ها ۴ نفره هستند. (از گروه بندی تمرین یک استفاده کنید.)**
- برای انجام این تمرین از ابزار Android Studio و زبان های برنامه نویسی Java یا Kotlin به انتخاب خود استفاده کنید .(استفاده از Qt و زبان C++ پیشنهاد نمیشود اما اگر گروهی خود میتواند دسترسی ها به ابزار را فراهم کند و مشکلات احتمالی نبود اینترنت بین الملل را به **طور مستقل** حل کند، مانعی وجود ندارد.)
- برنامه شما باید قابلیت اجرا بر Android 10 (API level 29) را داشته باشد.
- با توجه به قطعی Github یا GitLab ، برای مشاهده نقش هر یک از اعضا گروه در پیش برد پروژه، یک فایل شکست پروژه توسط هر گروه باید ساخته شود و نقش هر یک از اعضا شرح داده شود. به طور کلی سعی بر آن باشد که هر یک از اعضا نقش برابری داشته باشند.
- یک نفر از اعضای گروه کدها را به همراه توضیحات خواسته شده و جواب سوالات و فایل شکست پروژه را در صفحه درس با فرمت CPS-CA2-<SID1>-<SID2>-<SID3>-<SID4>.zip آپلود کند.
- علاوه بر کدها، فایل apk مربوط به پیاده‌سازی را نیز آپلود کنید. دقت کنید که فایل apk شما باید سازگار با اندرویدهای ۱۰ به بالا باشد.
- این تمرین احتمالا تحویل آنلاین دارد و از بخش ها مختلف از تمامی افراد سوال پرسیده می‌شود.
- یک ویدئو ضبط نمایید که در آن همزمان کارکرد موس بر روی صفحه لپتاپ و صفحه تلفن همراه قابل مشاهده باشد. طول ویدئو کوتاه(حداکثر ۵ دقیقه) باشد. چالش هایی که در طول انجام پروژه با آن برخورد کردید و اگر فرض خاص یا توضیح مهمی دارید را در ویدئو بیان نمایید.
- عدم رعایت مورد فوق، میتواند با کسر نمره همراه باشد.
- همه افراد باید به پروژه مسلط باشند و نمره تمامی اعضای گروه لزوما یکسان نخواهد بود.
- تمیزی و خوانایی کد و گزارش بخشی از نمره شما را تشکیل خواهد داد.
- این تمرین برای یادگیری شما طرح شده است. در صورت محرز شدن تقلب در تمرین، به طور جدی پیگیری خواهد شد.

موفق باشید