باسمه تعالی

دانشگاه صنعتی شریف
دانشکده مهندسی برق

رمزنگاری پیشرفته
اساتید درس: دکتر سلماسی زاده، دکتر احمدی
نیم‌سال دوم ۱۴۰۲ ـ ۱۴۰۱

پروژه نهایی درس

«نکات مهم»

- پروژه‌ها بصورت انفرادی است و هیچ دو فردی مجاز به انتخاب موضوع یکسان نیستند.
- اعلام موضوع انتخابی توسط شما از طریق ایمیل به mirzaie.atiyeh@yahoo.com خواهد بود و اولویت انتخاب هر پروژه با فردی است که زودتر موضوع خود را اطلاع داده. پس از دریافت ایمیل تایید، امکان شروع کار بر روی پروژه را خواهید داشت و در صورت تکمیل ظرفیت پروژه به شما اطلاع داده خواهد شد تا موضوع دیگری را انتخاب کنید.
- مهلت اعلام موضوع پروژه درس تا تاریخ ۳۰ فروردین خواهد بود.
- هر پروژه دارای دو بخش تئوری و عملی خواهد بود که لازم است در فاز اول تسلط مناسبی بر بخش تئوری صورت گیرد و تا تاریخ ۳۰ اردیبهشت گزارش فاز اول بارگذاری شود. در این گزارش حداکثر در **یک صفحه** آنچه که تا آن زمان مطالعه شده (نه جزئیات علمی، بلکه صرفا موضوعات مورد بررسی از هر مرجع) را بیان کنید. اگر احیانا با چالش‌هایی نیز مواجه هستید، آنها را نام ببرید و همچنین برنامه پیشروی خود را برای انجام پروژه ذکر کنید.
- در فاز دوم، هدف استفاده از مفاهیم تئوری فاز اول برای پیاده سازی بخش عملی پروژه است. در تعدادی از پروژه‌ها لازم است برنامه نویسی به زبان پایتون انجام شود که در کنار عنوان این پروژه‌ها ستاره قرمز رنگ قرار داده شده است (نداشتن بخش برنامه نویسی در سایر پروژه‌ها به معنای آسانتر بودن آنها نیست!). در پایان این فاز لازم است گزارش کاملی از بخش تئوری و عملی و شرح دقیق نتایج حاصل، به همراه کد پیاده‌سازی شده (برای پروژه‌های دارای برنامه نویسی) ارائه شود. نگارش گزارش تنها در قالبی که به پیوست آمده قابل قبول است. مهلت انجام این فاز متعاقبا اعلام خواهد شد.
- در پایان نیز ارائه‌ای خواهید داشت که تاریخ دقیق آن متعاقبا اعلام خواهد شد. لازم است فایل ارائه تا یک روز قبل آن بارگذاری شود. قالب‌های پیشنهادی ارائه نیز به پیوست آمده است.
- هر گونه ابهام و سوال خود را از طریق ایمیل mirzaie.atiyeh@yahoo.com یا آیدی @A_N_Mir در تلگرام یا بله بیان کنید و یا اشکال خود را در گروه تلگرامی درس مطرح کنید.

با آرزوی موفقیت

فهرست مطالب

پروژه ۱ *بازیابی کلید به روش مدل گرافی

بخش تئوری

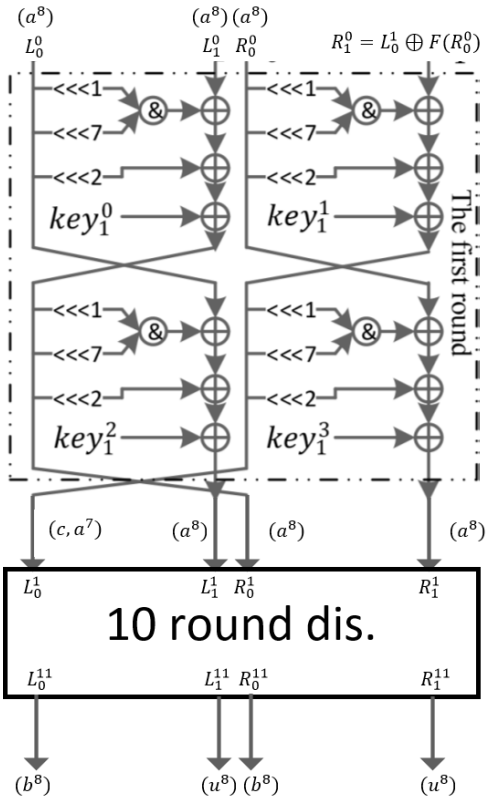
در این پروژه روش مدل گرافی برای بازیابی کلید را فرا می‌گیریم. بدین منظور لازم است ابتدا مقاله شماره ۱ در پوشه Refrences را مطالعه کنید. دقت شود که مطالب دیگری برای بهبود تمایزگر انتگرالی در این مقاله آورده شده که لازم نیست جزئیات آن‌ها مورد بررسی قرار گیرد و هدف، یادگیری روش بازیابی کلید است. همچنین لازم است جزییات ساختار رمز SHADOW۳۲ که در مقاله شماره ۲ آمده را مطالعه کنید. یادگیری منطق طراحی این رمز لازم نیست و تنها جزئیات ساختار رمز اهمیت دارد.

بخش عملی

فرض کنید تمایزگر ۱۱ دوری با ۸ بیت بالانس مطابق شکل ۱ برای رمز SHADOW۳۲ داریم (بیت‌هایی که در خروجی تمایزگر با b نشان داده شده‌اند همان بیت‌های بالانس هستند). حال به کمک روش بازیابی کلید مطالعه شده، بهترین حمله از نظر تعداد دور و سایر پیچیدگی‌های حمله را به SHADOW۳۲ اعمال کنید. پیاده سازی این بخش باید به زبان پایتون صورت گیرد. به عنوان راهنمایی می‌توانید از کدهای key-recovery در لینک زیر که مربوط به مقاله ۱ هستند کمک بگیرید.

https://github.com/hadipourh/mpt

امتیازی: همانطور که در مقاله ۴ ذکر شد، تعداد دور رمز SHADOW۳۲ ۱۶ دور است، اگر بتوانید به کمک روش مطالعه شده حمله‌ای ۱۶ دوری ارائه دهید که دارای پیچیدگی‌های مناسبی باشد نمره تشویقی لحاظ می‌شود.



شکل ۱: تمایزگر ۱۱ دوری SHADOW۳۲

پروژه ۲ تکنیک partial-sum و تمایزگر مستقیم و وارون

بخش تئوری

در این پروژه دو روش بهبود حمله انتگرالی را فرا میگیرید: روش patial-sum در مرحله بازیابی کلید و روش استفاده از تمایزگر وارون و مستقیم. بدین منظور لازم است ابتدا مقاله شماره ۳ در پوشه Refrences را مطالعه کنید. دقت شود که روش‌های دیگری نیز برای بهبود بازیابی کلید (مانند meet-in-the-middle و...) در این مقاله استفاده شده که لازم نیست جزئیات آن‌ها مورد بررسی قرار گیرد. همچنین لازم است جزییات ساختار رمز SAND۶۴ و حمله انتگرالی که در مقاله شماره ۴ آمده را مطالعه کنید. یادگیری نحوه به دست آوردن تمایزگر انتگرالی لازم نیست و تنها خود تمایزگر و نحوه استفاده از آن در مرحله بازیابی کلید اهمیت دارد.

بخش عملی

در این بخش باید به کمک حداقل یکی از دو تکنیک مطالعه شده در بخش تئوری، حمله انتگرالی در مقاله شماره ۴ را بهبود ببخشید. برای این کار لازم است ابتدا نشان دهید به دلیل ساختار رمز، تمایزگر انتگرالی وارون (در جهت رمزگشایی) برای SAND۶۴ وجود دارد و از روی تمایزگر مستقیم مقاله ۴ به دست می‌آید. سپس این تمایزگر را به دست آورید و مشابه روش مقاله ۳ به روشی کارا از هر دو تمایزگر در فرآیند حمله استفاده کنید. دقت کنید تنها در صورتی پیاده‌سازی شما صحیح خواهد بود که نتایج نهایی حمله شما نسبت به آنچه در مقاله شماره ۴ آمده بهتر باشد؛ یعنی حداقل یکی از پیچیدگی‌های حمله کاهش یا تعداد دور حمله افزایش یافته باشد. همچنین می‌توانید به جای اعمال روش تمایزگر مستقیم و وارون، روش partial-sum را در مرحله بازیابی کلید مقاله ۴ اعمال کنید و نتایج را بهبود ببخشید.

امتیازی: چنانچه بتوانید با استفاده از این دو تکنیک و یا ترکیب آن با هر روش دیگری، حمله مقاله شماره ۴ را بهبود چشمگیری ببخشید برای شما نمره تشویقی لحاظ می‌شود.

پروژه ۳ تکنیک meet-in-the-middle در بازیابی کلید

بخش تئوری

در این پروژه قصد داریم با بهره‌گیری از meet-in-the-middle مرحله بازیابی کلید در حمله انتگرالی را بهبود ببخشیم. بدین منظور لازم است ابتدا مقاله شماره ۳ در پوشه Refrences را مطالعه کنید. دقت شود که روش‌های دیگری نیز برای بهبود بازیابی کلید (مانند partial-sum و...) در این مقاله استفاده شده که لازم نیست جزئیات آن‌ها مورد بررسی قرار گیرد؛ تمرکز پروژه فعلی بر روی تکنیک meet-in-the-middle است. همچنین لازم است جزییات ساختار رمز SAND۶۴ و مرحله بازیابی کلید حمله انتگرالی که در مقاله شماره ۴ آمده را مطالعه کنید. یادگیری نحوه به دست آوردن تمایزگر انتگرالی لازم نیست و تنها خود تمایزگر و نحوه استفاده از آن در مرحله بازیابی کلید اهمیت دارد.

بخش عملی

در این بخش باید به کمک تکنیک مطالعه شده در بخش تئوری، حمله انتگرالی در مقاله شماره ۴ را بهبود ببخشید. برای این کار لازم است مشابه مقاله ۳، ابتدا معادله حمله را به کمک بیت‌های بالانس بنویسید و با بررسی جدول زیرکلیدهای مورد نیاز در بازیابی کلید، بیت‌هایی که مستقل از یکدیگر تاثیر میگذارند را مشخص کرده و تکنیک meet-in-the-middle را اعمال کنید. دقت کنید تنها در صورتی پیاده‌سازی شما صحیح خواهد بود که نتایج نهایی حمله شما نسبت به آنچه در مقاله شماره ۴ آمده‌اند بهتر باشند؛ یعنی حداقل یکی از پیچیدگی‌های حمله کاهش یا تعداد دور حمله افزایش یافته باشد.

امتیازی: چنانچه بتوانید با استفاده از این تکنیک و یا ترکیب آن با هر روش دیگری، حمله مقاله شماره ۴ را بهبود چشمگیری ببخشید برای شما نمره تشویقی لحاظ می‌شود.

پروژه ۴ *جستجوی خودکار تمایزگر تفاضل ناممکن

بخش تئوری

در این پروژه نحوه یافتن تمایزگر تفاضل ناممکن به روش جستجوی خودکار را فرا می‌گیریم. بدین منظور لازم است مقاله شماره ۵ در پوشه Refrences را مطالعه کنید. دقت کنید این مقاله مطالب دیگری ذکر کرده که لازم به مطالعه آنها نیست (مانند تمایزگر همبستگی صفر و انتگرالی و نحوه پیاده سازی خودکار بازیابی کلید برای تمامی این حملات). همچنین لازم است ساختار رمز Simon۳۲/۴۸ و بهترین تمایزگرهای تفاضل ناممکن موجود را بدانید که این موارد در مقاله شماره ۶ آورده شده‌اند. در این مقاله مطالب دیگری نیز آورده که نیاز به مطالعه آنها نیست. جدول ۱ در مقاله ۶ بطور خلاصه بهترین تمایزگرهای موجود برای Simon۳۲/۴۸ را نشان داده است.

بخش عملی

در بخش عملی پروژه، روشی که برای جستجوی خودکار تمایزگر تفاضل ناممکن فرا گرفته‌اید را پیاده‌سازی می‌کنید. پیاده سازی به زبان پایتون است و باید ابتدا کتابخانه gurobipy نصب شود. برای این کار ابتدا به کمک دستور زیر نسخه محدود آن را نصب کنید:

```
pip install gurobipy
```

سپس برای فعالسازی نسخه نامحدود آن به لینک زیر مراجعه کرده و پس از تشکیل حساب کاربری، لایسنس رایگان آکادمیک را فعال کنید. لازم به ذکر است برای فعالسازی این لایسنس باید حتما به vpn شریف خود وصل باشید.

```
https://www.gurobi.com/downloads/gurobi-software/
```

همچنین برای یادگیری نحوه بهینه سازی به کمک این کتابخانه میتوانید از لینک زیر کمک بگیرید:

```
https://www.gurobi.com/documentation/
```

حال به کمک روش مطالعه شده در مقاله ۵، بهترین تمایزگر تفاضل ناممکن را برای یکی از دو رمز Simon۳۲ یا Simon۴۸ بیابید. دقت کنید تنها در صورتی پیاده‌سازی شما صحیح خواهد بود که نتایج نهایی تمایزگر شما نسبت به آنچه در مقاله شماره ۶ آمده‌اند بهتر باشند؛ یعنی پیچیدگی داده کاهش یا تعداد دور تمایزگر افزایش یافته باشد.

امتیازی: چنانچه بتوانید با استفاده از این تکنیک و یا ترکیب آن با هر روش دیگری، تمایزگر مقاله شماره ۶ را بهبود چشمگیری ببخشید برای شما نمره تشویقی لحاظ می‌شود.

پروژه ۵ *بازیابی خودکار کلید در حمله تفاضل ناممکن

بخش تئوری

در این پروژه نحوه به دست آوردن بازیابی کلید بهینه برای حمله تفاضل ناممکن به روش جستجوی خودکار را فرا می‌گیریم. بدین منظور لازم است مقاله شماره ۵ در پوشه Refrences را مطالعه کنید. دقت کنید این مقاله مطالب دیگری ذکر کرده که لازم به مطالعه آنها نیست (مانند تمایزگر تفاضل ناممکن، همبستگی صفر و انتگرالی و نحوه پیاده سازی خودکار بازیابی کلید برای حملات دیگر). همچنین لازم است ساختار رمز Simon۳۲/۴۸ و بهترین تمایزگرهای تفاضل ناممکن موجود را بدانید که این موارد در مقاله شماره ۶ آورده شده‌اند. در این مقاله مطالب دیگری نیز آورده که نیاز به مطالعه آنها نیست. جدول ۱ در مقاله ۶ بطور خلاصه بهترین تمایزگرهای موجود برای Simon۳۲/۴۸ را نشان داده است.

بخش عملی

در این بخش از پروژه، روشی که برای جستجوی خودکار بازیابی کلید در حمله تفاضل ناممکن فرا گرفته‌اید را پیاده‌سازی می‌کنید. پیاده سازی به زبان پایتون است و باید ابتدا کتابخانه gurobipy نصب شود. نحوه نصب این کتابخانه در پروژه ۴ آورده شده است.

حال به کمک روش مطالعه شده در مقاله ۵، بهترین حمله تفاضل ناممکن را برای یکی از دو رمز Simon۳۲ یا Simon۴۸ بیابید. توجه شود که پیاده سازی بخش تمایزگر با شما نیست و تنها باید از تمایزگر مربوطه در مقاله ۶ استفاده کنید و باقی حمله (بخش بازیابی کلید) را پیاده کنید. دقت کنید تنها در صورتی پیاده‌سازی شما صحیح خواهد بود که نتایج نهایی حمله شما نسبت به آنچه در مقاله شماره ۶ آمده‌اند بهتر باشند؛ یعنی پیچیدگی زمانی کاهش یا تعداد دور حمله افزایش یافته باشد.

امتیازی: چنانچه بتوانید با استفاده از این تکنیک و یا ترکیب آن با هر روش دیگری، حمله مقاله شماره ۶ را بهبود چشمگیری ببخشید برای شما نمره تشویقی لحاظ می‌شود.

پروژه ۶ *جستجوی خودکار تمایزگر همبستگی صفر

بخش تئوری

در این پروژه نحوه یافتن تمایزگر همبستگی صفر به روش جستجوی خودکار را فرا می‌گیریم. بدین منظور لازم است مقاله شماره ۵ در پوشه Refrences را مطالعه کنید. دقت کنید این مقاله مطالب دیگری ذکر کرده که لازم به مطالعه آنها نیست (مانند نحوه پیاده سازی خودکار بازیابی کلید). همچنین لازم است ساختار رمز Simon۳۲/۴۸ و بهترین تمایزگرهای همبستگی صفر موجود را بدانید که این موارد در مقاله شماره ۶ آورده شده‌اند. در این مقاله مطالب دیگری نیز آورده که نیاز به مطالعه آنها نیست. جدول ۱ در مقاله ۶ بطور خلاصه بهترین تمایزگرهای موجود برای Simon۳۲/۴۸ را نشان داده است.

بخش عملی

در بخش عملی پروژه، روشی که برای جستجوی خودکار تمایزگر همبستگی صفر فرا گرفته‌اید را پیاده‌سازی می‌کنید. پیاده سازی به زبان پایتون است و باید ابتدا کتابخانه gurobipy نصب شود. نحوه نصب این کتابخانه در پروژه ۴ آورده شده است.

حال به کمک روش مطالعه شده در مقاله ۵، بهترین تمایزگر همبستگی صفر را برای یکی از دو رمز Simon۳۲ یا Simon۴۸ بیابید. دقت کنید تنها در صورتی پیاده‌سازی شما صحیح خواهد بود که نتایج نهایی تمایزگر شما نسبت به آنچه در مقاله شماره ۶ آمده‌اند بهتر باشند؛ یعنی پیچیدگی داده کاهش یا تعداد دور تمایزگر افزایش یافته باشد.

امتیازی: چنانچه بتوانید با استفاده از این تکنیک و یا ترکیب آن با هر روش دیگری، تمایزگر مقاله شماره ۶ را بهبود چشمگیری ببخشید برای شما نمره تشویقی لحاظ می‌شود.

پروژه ۷ *بازیابی خودکار کلید در حمله همبستگی صفر

بخش تئوری

در این پروژه نحوه به دست آوردن بازیابی کلید بهینه برای حمله همبستگی صفر به روش جستجوی خودکار را فرا می‌گیریم. بدین منظور لازم است مقاله شماره ۵ در پوشه Refrences را مطالعه کنید. دقت کنید این مقاله مطالب دیگری ذکر کرده که لازم به مطالعه آنها نیست (مانند جستجوی خودکار تمایزگر تفاضل ناممکن، همبستگی صفر و انتگرالی). همچنین لازم است ساختار رمز Simon۳۲/۴۸ و بهترین تمایزگرهای همبستگی صفر موجود را بدانید که این موارد در مقاله شماره ۶ آورده شده‌اند. در این مقاله مطالب دیگری نیز آورده که نیاز به مطالعه آنها نیست. جدول ۱ در مقاله ۶ بطور خلاصه بهترین تمایزگرهای موجود برای Simon۳۲/۴۸ را نشان داده است.

بخش عملی

در این بخش از پروژه، روشی که برای جستجوی خودکار بازیابی کلید در حمله همبستگی صفر فرا گرفته‌اید را پیاده‌سازی می‌کنید. پیاده سازی به زبان پایتون است و باید ابتدا کتابخانه gurobipy نصب شود. نحوه نصب این کتابخانه در پروژه ۴ آورده شده است.

حال به کمک روش مطالعه شده در مقاله ۵، بهترین حمله همبستگی صفر را برای یکی از دو رمز Simon۳۲ یا Simon۴۸ بیابید. توجه شود که پیاده سازی بخش تمایزگر با شما نیست و تنها باید از تمایزگر مربوطه در مقاله ۶ استفاده کنید و باقی حمله (بخش بازیابی کلید) را پیاده کنید. دقت کنید تنها در صورتی پیاده‌سازی شما صحیح خواهد بود که نتایج نهایی حمله شما نسبت به آنچه در مقاله شماره ۶ آمده‌اند بهتر باشند؛ یعنی پیچیدگی زمانی کاهش یا تعداد دور حمله افزایش یافته باشد.

امتیازی: چنانچه بتوانید با استفاده از این تکنیک و یا ترکیب آن با هر روش دیگری، حمله مقاله شماره ۶ را بهبود چشمگیری ببخشید برای شما نمره تشویقی لحاظ می‌شود.

پروژه ۸ *جستجوی خودکار تمایزگر انتگرالی

بخش تئوری

در این پروژه نحوه یافتن تمایزگر انتگرالی به روش جستجوی خودکار را فرا می‌گیریم. بدین منظور لازم است مقاله شماره ۵ در پوشه Refrences را مطالعه کنید. دقت کنید این مقاله مطالب دیگری ذکر کرده که لازم به مطالعه آنها نیست (مانند نحوه پیاده سازی خودکار بازیابی کلید). همچنین لازم است ساختار رمز Simon۳۲/۴۸ و بهترین تمایزگرهای انتگرالی موجود را بدانید که این موارد در مقاله شماره ۷ آورده شده‌اند. در این مقاله مطالب دیگری نیز آورده که نیاز به مطالعه آنها نیست. جدول ۱ در مقاله ۷ بطور خلاصه بهترین تمایزگرهای موجود برای Simon۳۲/۴۸ را نشان داده است.

بخش عملی

در بخش عملی پروژه، روشی که برای جستجوی خودکار تمایزگر انتگرالی فرا گرفته‌اید را پیاده‌سازی می‌کنید. پیاده سازی به زبان پایتون است و باید ابتدا کتابخانه gurobipy نصب شود. نحوه نصب این کتابخانه در پروژه ۴ آورده شده است.

حال به کمک روش مطالعه شده در مقاله ۵، بهترین تمایزگر انتگرالی را برای یکی از دو رمز Simon۳۲ یا Simon۴۸ بیابید. دقت کنید تنها در صورتی پیاده‌سازی شما صحیح خواهد بود که نتایج نهایی تمایزگر شما نسبت به آنچه در مقاله شماره ۷ آمده‌اند بهتر باشند؛ یعنی پیچیدگی داده کاهش یا تعداد دور تمایزگر افزایش یافته باشد.

امتیازی: چنانچه بتوانید با استفاده از این تکنیک و یا ترکیب آن با هر روش دیگری، تمایزگر مقاله شماره ۷ را بهبود چشمگیری ببخشید برای شما نمره تشویقی لحاظ می‌شود.

پروژه ۹ *بازیابی خودکار کلید در حمله انتگرالی

بخش تئوری

در این پروژه نحوه به دست آوردن بازیابی کلید بهینه برای حمله انتگرالی به روش جستجوی خودکار را فرا می‌گیریم. بدین منظور لازم است مقاله شماره ۵ در پوشه Refrences را مطالعه کنید. دقت کنید این مقاله مطالب دیگری ذکر کرده که لازم به مطالعه آنها نیست (مانند جستجوی خودکار تمایزگر تفاضل ناممکن، همبستگی صفر و انتگرالی). همچنین لازم است ساختار رمز Simon۳۲/۴۸ و بهترین تمایزگرها و حملات انتگرالی موجود را بدانید که این موارد در مقاله شماره ۶ و ۷ آورده شده‌اند. در این دو مقاله مطالب دیگری نیز آورده که نیاز به مطالعه آنها نیست. بهترین تمایزگرهای انتگرالی برای این دو رمز در جدول ۱ مقاله ۷ و بهترین حملات انتگرالی برای این دو رمز در جدول ۱ مقاله ۶ بطور خلاصه ذکر شده است.

بخش عملی

در این بخش از پروژه، روشی که برای جستجوی خودکار بازیابی کلید در حمله انتگرالی را فرا گرفته‌اید را پیاده‌سازی می‌کنید. پیاده سازی به زبان پایتون است و باید ابتدا کتابخانه gurobipy نصب شود. نحوه نصب این کتابخانه در پروژه ۴ آورده شده است.

حال به کمک روش مطالعه شده در مقاله ۵، بهترین حمله انتگرالی را برای یکی از دو رمز Simon۳۲ یا Simon۴۸ بیابید. توجه شود که پیاده سازی بخش تمایزگر با شما نیست و تنها باید از تمایزگر مربوطه در مقاله ۷ استفاده کنید و باقی حمله (بخش بازیابی کلید) را پیاده کنید. دقت کنید تنها در صورتی پیاده‌سازی شما صحیح خواهد بود که نتایج نهایی حمله شما نسبت به آنچه در مقاله شماره ۶ آمده‌اند بهتر باشند؛ یعنی پیچیدگی زمانی کاهش یا تعداد دور حمله افزایش یافته باشد.

امتیازی: چنانچه بتوانید با استفاده از این تکنیک و یا ترکیب آن با هر روش دیگری، حمله مقاله شماره ۶ را بهبود چشمگیری ببخشید برای شما نمره تشویقی لحاظ می‌شود.

پروژه ۱۰ *جستجوی خودکار تمایزگر انتگرالی با استفاده از ویژگی تقسیم

بخش تئوری

در این پروژه نحوه یافتن تمایزگر انتگرالی به روش جستجوی خودکار با استفاده از ویژگی تقسیم را فرا می‌گیریم. بدین منظور لازم است مقاله شماره ۸ در پوشه Refrences را به دقت مطالعه کنید. همچنین لازم است جزییات ساختار رمز DABC که در مقاله شماره ۹ آمده را مطالعه کنید. یادگیری منطق طراحی این رمز لازم نیست و تنها جزئیات ساختار رمز اهمیت دارد.

بخش عملی

در بخش عملی پروژه، روشی که برای جستجوی خودکار تمایزگر انتگرالی فرا گرفته‌اید را پیاده‌سازی می‌کنید. پیاده سازی به زبان پایتون است و باید ابتدا کتابخانه gurobipy نصب شود. نحوه نصب این کتابخانه در پروژه ۴ آورده شده است.
حال به کمک روش مطالعه شده در مقاله ۸، بهترین تمایزگر انتگرالی (یعنی تمایزگر با بیشترین تعداد دور و بیت بالانس و کمترین پیچیدگی داده) را برای رمز DABC بیابید. به عنوان راهنمایی می‌توانید از کدهای لینک زیر که مربوط به مقاله ۸ هستند کمک بگیرید.

`https://github.com/xiangzejun/MILP_Division_Property`

امتیازی: چنانچه بتوانید این روش را بر روی رمز دیگری که پس از سال ۲۰۱۹ ارائه شده (مانند SCENERY یا GFRX یا...) اعمال کنید و تمایزگر مناسبی به دست آورید برای شما نمره تشویقی لحاظ می‌شود.

پروژه ۱۱ *مدلسازی Sbox در انتشار ویژگی تقسیم

بخش تئوری

در این پروژه روش جدیدی برای مدلسازی Sbox به عنوان بخشی از قیود مساله بهینه سازی در یافتن خودکار تمایزگر انتگرالی را فرا می‌گیریم. بدین منظور لازم است ابتدا مقاله شماره ۸ در پوشه Refrences را مطالعه کنید تا با مفهوم کلی ویژگی تقسیم و مساله بهینه سازی در جستجوی خودکار تمایزگر انتگرالی آشنا شوید. حال لازم است روش مدلسازی Sbox در مقاله ۷ را به دقت مطالعه کنید. توجه شود که مطالب دیگری نیز در این مقاله ذکر شده که نیاز به مطالعه آنها نیست.

بخش عملی

در بخش عملی پروژه، روشی که برای مدلسازی Sbox در مقاله ۷ فرا گرفته‌اید را پیاده‌سازی می‌کنید و با روش‌های متداول مقایسه می‌کنید. برای پیاده سازی یکی از روشهای متداول ابتدا دنباله‌های تقسیم معتبر را با اجرا کردن کد SboxCharec-tristic.py که به پیوست آمده برای Sbox سه رمز LED، CRAFT و TWINE به دست آورید. دقت کنید رابطه Sbox هر سه رمز داخل کد آورده شده و صرفا با uncomment کردن خط متناظر با همان رمز می‌توان دنباله‌های تقسیم معتبر متناظرش را به دست آورد. سپس به کمک نرم افزار SageMath نامساوی‌های توصیف کننده هر سه Sbox را به دست آورید. لینک نصب این نرم افزار در زیر آورده شده است:

`https://github.com/sagemath/sage-windows/releases`

حال به عنوان روش دوم به کمک تکنیک مطالعه شده در مقاله ۷، دوباره نامساوی‌های توصیف کننده هر Sbox را به دست آورید (پیاده سازی با پایتون). در نهایت تعداد نامساوی‌های توصیف کننده دو روش را در جدولی گزارش کنید و بیان کنید کدام روش بهتر بوده است (طبیعتا روشی که بتواند هر Sbox را با تعداد کمتری نامساوی توصیف کند روش بهتری است).

امتیازی: چنانچه بتوانید این روش را بر روی حداقل ۲ رمز دیگری که پس از سال ۲۰۱۹ ارائه شده اعمال کنید و به نتیجه بهتری نسبت به روش استفاده از SageMath برسید برای شما نمره تشویقی لحاظ می‌شود.

پروژه ۱۲ *مدلسازی لایه‌ی خطی در انتشار ویژگی تقسیم

بخش تئوری

در این پروژه روش جدیدی برای مدلسازی لایه خطی به عنوان بخشی از قیود مساله بهینه سازی در یافتن خودکار تمایزگر انتگرالی را فرا می‌گیریم. بدین منظور لازم است مقاله ۱۰ و ۱۱ در پوشه Refrences را مطالعه کنید. توجه کنید که دقت پروژه بر روی مدلسازی لایه خطی و کلیت مفهوم ویژگی تقسیم و مساله بهینه سازی در جستجوی خودکار تمایزگر انتگرالی است و لازم نیست جزئیات مطالب دیگری که در دو مقاله بیان شده مطالعه شود.

بخش عملی

در بخش عملی پروژه، روشی که برای مدلسازی لایه خطی در مقاله ۱۰ فرا گرفته‌اید را پیاده‌سازی می‌کنید و با نتایج روش مقاله ۱۱ مقایسه می‌کنید. برای این منظور روش مقاله ۱۰ را به زبان پایتون پیاده کنید و نامساوی‌های توصیف کننده لایه خطی دو رمز SKINNY و Midori را به دست آورید (ماتریس لایه خطی هر دو رمز در مقاله ۱۱ آورده شده است). حال تعداد نامساوی‌های توصیف کننده دو روش را در جدولی گزارش کنید و بیان کنید کدام روش بهتر بوده است (طبیعتا روشی که بتواند هر Sbox را با تعداد کمتری نامساوی توصیف کند روش بهتری است).

امتیازی: چنانچه بتوانید این روش را بر روی حداقل ۲ رمز دیگری که پس از سال ۲۰۱۹ ارائه شده اعمال کنید و به نتیجه بهتری نسبت به روش مقاله ۱۱ برسید برای شما نمره تشویقی لحاظ می‌شود.