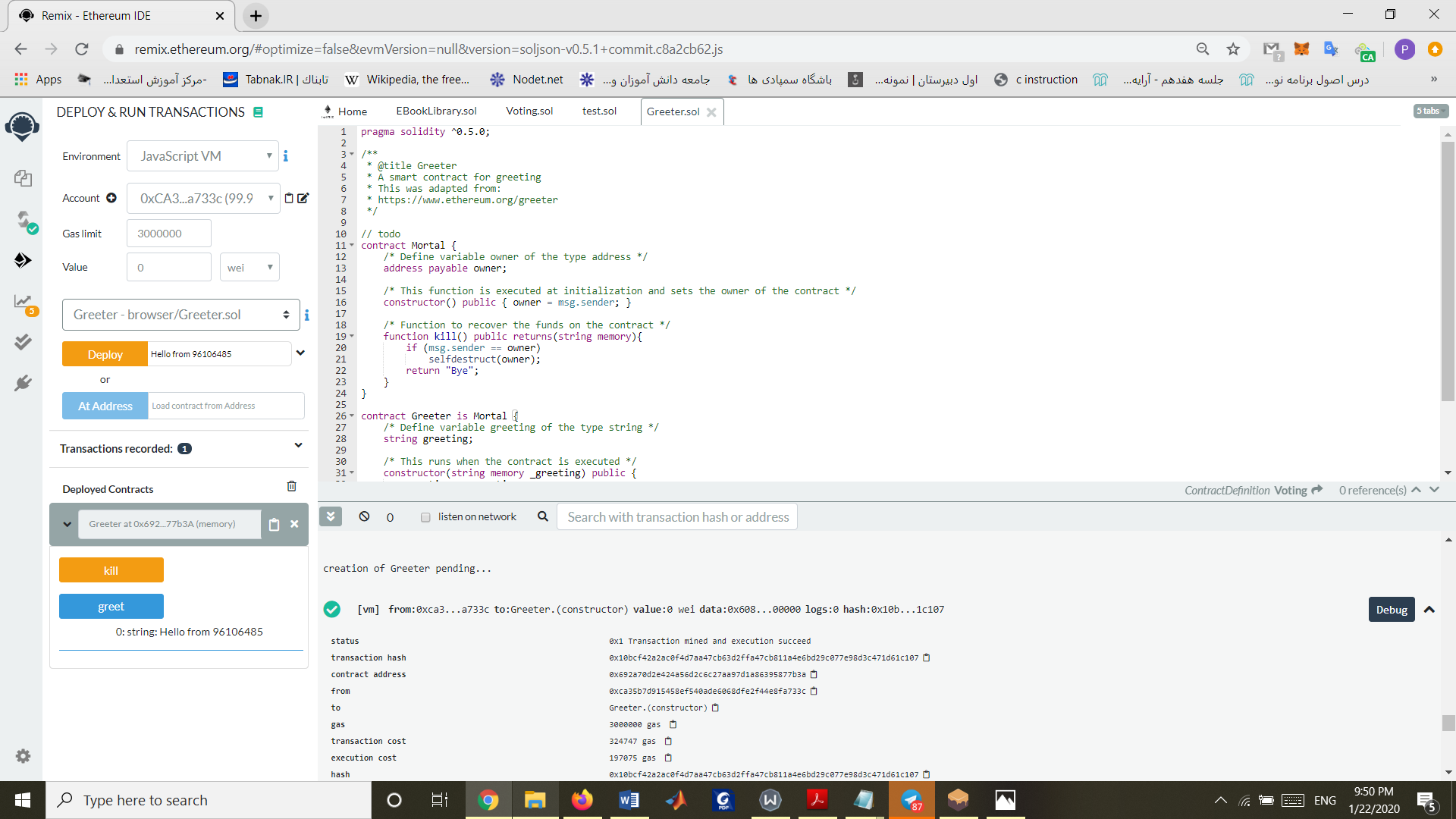
باسمه تعالی

گزارش تمرین عملی سری سوم پوریا دادخواه 96106485

1. deploy شده ی contract :



2.

توابع و متغیر های معمول ERC20 را از فایل github استفاده کردم.

برای توابع و event های مربوط به pause و time expire ها هم هر کدام جداگانه در کد کامنت گذاری شده که چه کاری لازم است انجام شود.

3 .

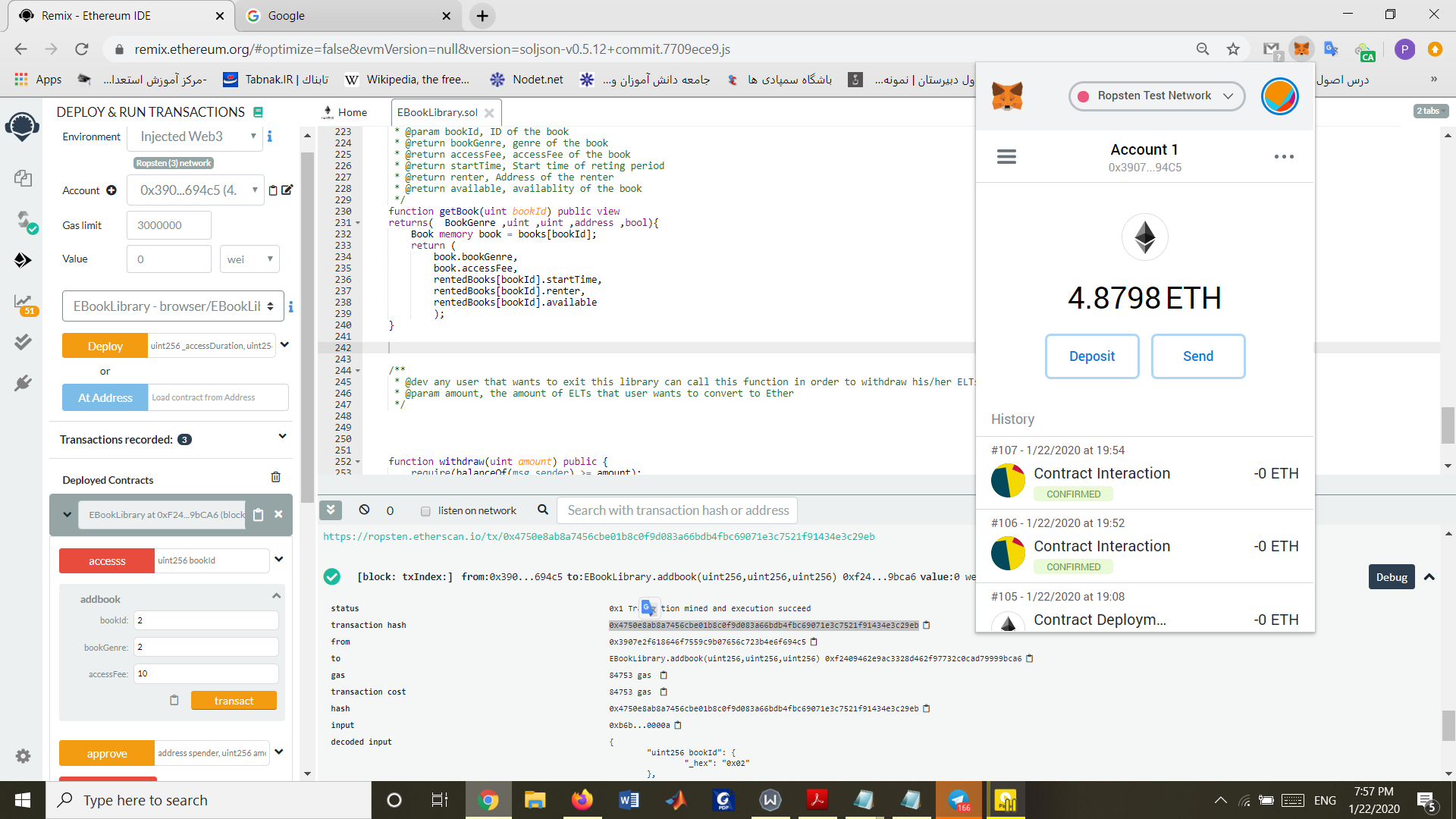
\*برای تابع addbook کافیست valid بودن کتاب را true کرده و مقادیر bookGenere و accessfee را ست کنیم.

\*برای تابع change to premium باید با استفاده از \_transfer که درERC20 تعریف کردیم صدا زننده تابع و کانترکت را admin کنیم و طبیعتا fee مربوط را نیز به عنوان ورودی میدهیم.

حال account type او را به نوع premium ست می کنیم.

\*برای revoke هم available بودن کتاب را true میکنیم.

4.



Book (1 0 3)

Contract address1 : [0xf2409462e9ac3328d462f97732c0cad79999bca6](https://ropsten.etherscan.io/address/0xf2409462e9ac3328d462f97732c0cad79999bca6)

Tx Hash1 : 0xb86458052586382e0a91c947fc7d86576fe2892fac8543193df356eda090734a

Book(2 2 10)

Contract address2 : [0xf2409462e9ac3328d462f97732c0cad79999bca6](https://ropsten.etherscan.io/address/0xf2409462e9ac3328d462f97732c0cad79999bca6)

Tx Hash2 : 0x4750e8ab8a7456cbe01b8c0f9d083a66bdb4fbc69071e3c7521f91434e3c29eb

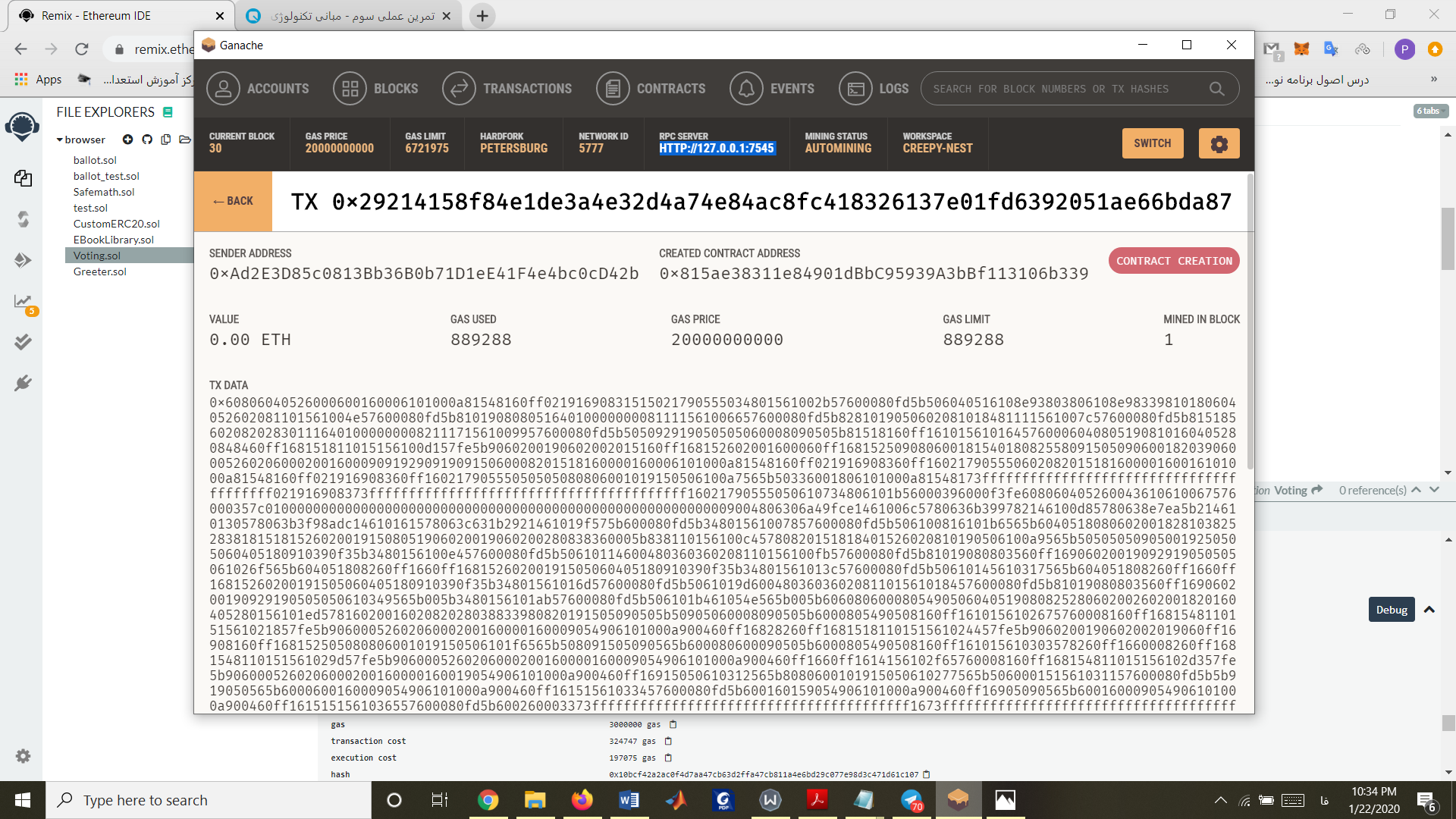
5.

a) یک اشکال اصلی این رای گیری این است که در این vote ما هیچ پروسه proof of work ویا proof of stake یا .. نداریم و از طرفی ساخت account0020 های مختلف هیچ زحمتی ندارد بنابرین میتوان با account های مختلف چندین بار رای داد و عملا رای گیری بی ارزش می شود...

برای حل آن هم میتوان حداقل یک کار محاسباتی ساده برای deploy کردن رای در نطر بگیریم. مثلا میتوان یک تابع در کانترکت قرار دارد که صرفا بدون گرفتن هیچ ورودی از کاربر هربار که اجرا شود عدد رندوم تولید کند و هدف این باشد که عدد تولیدی از یک ترشولدی بزرگتر باشد....

6.

پس از نصب و انجام عملیات های گفته شده voting contract را از اولین آدرس از 10 آدرس ganache ،با 9 کاندید با اندیس های 1تا9 deploy کردم:



و نهایتا UI تولید شده پس از رای گیری با اکانت های مختلف داریم:

