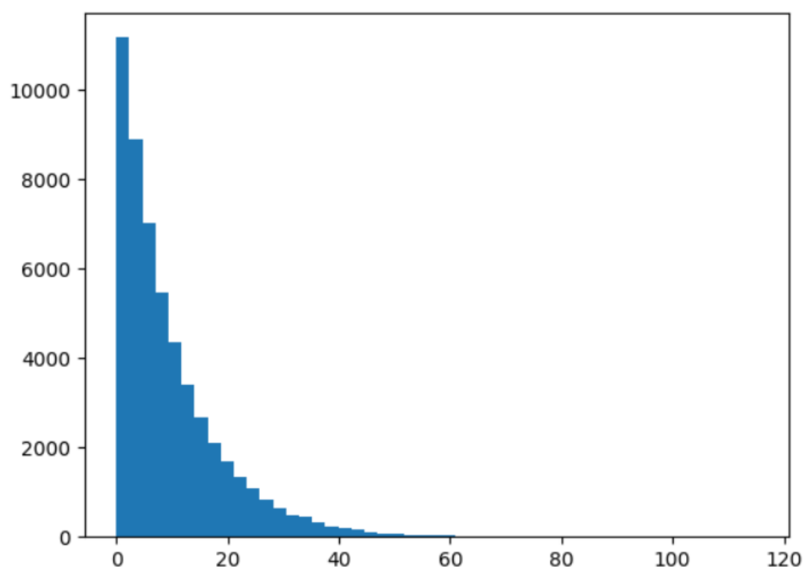
سوال اول

۱. استخراج بلاکچین را می‌توانیم یک رخداد برنولی در نظر بگیریم که با احتمال p رخ می‌دهد(که وابسته به difficulity است) که با ریت ثابتی (HashRate) آزمایش می‌شود. با توجه به این موضوع، همان‌طور که در کلاس نیز گفته شد، می‌توان تعداد رخدادهای برنولی در یک بازه زمانی را با استفاده از توزیع پوآسون مدل کرد. از طرفی می‌دانیم که فاصله زمانی میان دو رخداد پوآسون نیز از توزیع نمایی می‌باشد. بنابراین توزیع نمایی مدل مناسبی برای زمان بین استخراج بلاک‌ها می‌باشد. اگر فرض کنیم که نرخ استخراج بلاک در یک واحد زمانی برابر λ می‌باشد، آن‌گاه طبق توزیع نمایی، میانگین و واریانس فاصله زمانی میان استخراج دو بلاک به ترتیب برابر $\frac{1}{\lambda}$ و $\frac{1}{\lambda^2}$ می‌باشد.

۲. برای دریافت اطلاعات بلاک‌ها از سایت گفته شده یک کد پایتون نوشته شده است که همراه با فایل تمرین ضمیمه شده است. در این کد ابتدا فایل‌های مربوط به سال ۲۰۲۲ از سایت گرفته شده و در یک dataframe ذخیره می‌شوند. سپس زمان میانی استخراج بلاک‌ها را از داده‌های داده شده استخراج کرده و هیستوگرام و میانگین و واریانس آن به دست می‌آید. همچنین از تست Kolmogorov-Smirnov برای بررسی آن‌که داده‌ها از توزیع نمایی هستند، استفاده شده است. نتایج این تست‌ها در زیر قابل مشاهده است:




```
mean: 9.881480123589098
variance: 98.29269688085358
Accepted
```

همان‌طور که مشاهده می‌شود میانگین و واریانس زمان میان استخراج دو بلاک به ترتیب تقریبا برابر 10 و 100 می‌باشد که با انتظاری که از آن داریم(توزیع نمایی با میانگین 10) هم‌خوانی دارد. همچنین نمودار هیستوگرام و آزمون KS نیز نمایی بودن این توزیع را تایید می‌کنند.

کد استخراج داده:

```
def read_data():
  start = date(2022, 1, 1)
  end = date(2022, 12, 31)
  delta = timedelta(days=1)
  data = []
  while(start <= end):
    url =
'https://gz.blockchair.com/bitcoin/blocks/blockchair_bitcoin_blocks_' +
start.strftime('%Y%m%d') + '.tsv.gz'
    req = requests.get(url, allow_redirects=True)
    open('file.gz', 'wb').write(req.content)
    data.append(pd.read_csv('file.gz', sep='\t', compression='gzip'))
    start += delta
  return pd.concat(data)
```

کد تست KS:

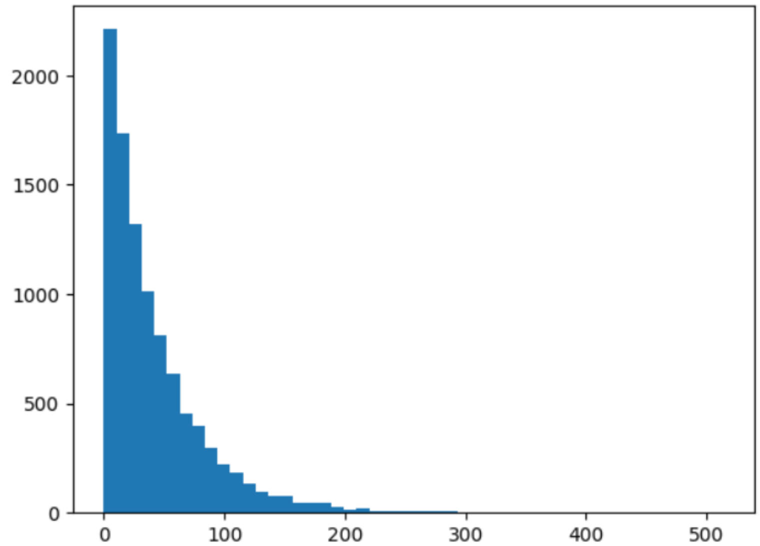
```
def test(blocks):
  durations = get_durations(blocks)
  bins = np.linspace(0, np.max(durations), 50)
  plt.hist(durations, bins=bins)
  plt.show()
  mean = np.mean(durations)
  var = np.var(durations)
  print('mean:', mean)
  print('variance:', var)

  alpha = 0.05
  null_hypothesis = lambda x: expon.cdf(x, scale=mean)
  test_statistic, p_value = kstest(durations, null_hypothesis)
  if p_value >= alpha:
    print('Accepted')
  else:
    print('Rejected')
```

۳. با مراجعه به سایت گفته شده می‌توانیم نسبت HashRate ماینرهای برتر را به کل HashRate شبکه ببینیم:

دو ماینر برتر این نمودار که در حقیقت دوتا از poolهای مشهور می‌باشند را در نظر می‌گیریم. Foundry USA Pool و AntPool به ترتیب 30% و 23% از توان پردازشی کل شبکه را دارند. بنابراین اگر نرخ استخراج کل شبکه را براب 0.1(یک بلاک در هر 10 دقیقه) در نظر بگیریم، این دو pool به ترتیب نرخی برابر 0.03 و 0.023 دارند، که یعنی زمان میان استخراج دو بلاک برای آن‌ها از توزیع‌های نمایی با میانگین‌های $\frac{1}{0.03} = 33.33$ و $\frac{1}{0.023} = 43.47$ پیروی می‌کند.

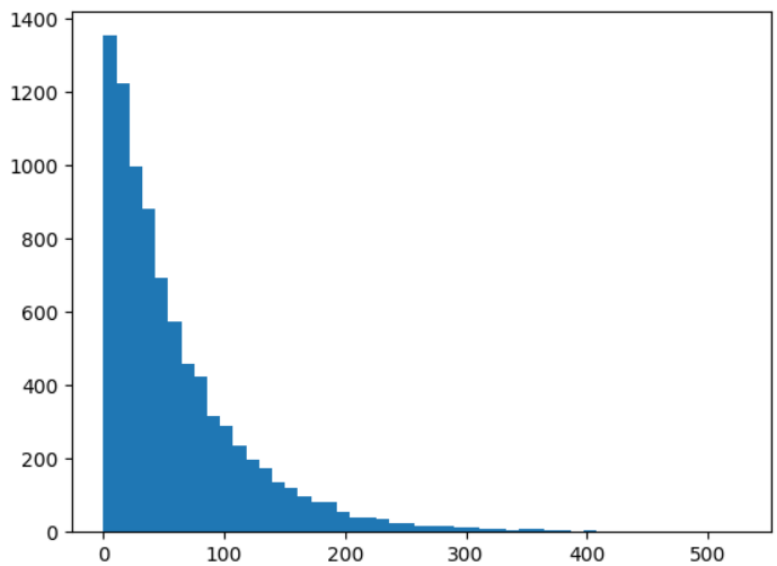
حال اگر تست قسمت قبل را برروی بلاک‌های استخراج این ماینرها انجام دهیم به نتایج زیر می‌رسیم:

```
test(df[df['guessed_miner'] == 'Foundry USA Pool'])
```




```
mean: 42.13254961318533
variance: 1881.7786886446681
Accepted
```

```
[ ] test(df[df['guessed_miner'] == 'AntPool'])
```




```
mean: 60.53538978494624
variance: 3683.239711304801
Accepted
```

همانطور که مشاهده می‌شوند در هر دوحالت تست‌های آماری نمایی بودن توزیع‌ها را تایید می‌کنند. میانگین و واریانس‌های به دست می‌آید.dsd

سوال دوم

۱. فرض کنیم که حد آستانه برابر T باشد. از آنجا که بیت‌کوین به صورت میانگین در هر ۱۰ دقیقه استخراج می‌شود پس داریم:

$$\lambda = 204.9 \times 10^{18} \times \frac{T}{2^{256}} = \frac{1}{60 \times 10}$$

$$\Rightarrow T = \frac{2^{256}}{204.9 \times 10^{18} \times 600} \cong 9.418 \times 10\text{^}53$$

می‌دانیم که آستانه‌ی صحیح در بیت‌کوین باید توانی از دو باشد، بنابراین برای محاسبه آن باید از T بالا لگارتیم مبنای ۲ بگیریم:

$$T_{correct} = 2^{\log_2 T} \approx 2^{179}$$

علت تفاوت این دو مقدار آن است که آستانه دقیق که به دست می‌آید خیلی اوقات توانی از دو نمی‌باشد.

۲.

$$E = \frac{204.9 \times 10^{18} \times 24}{95 \times 10^{12}} \times 3250 \approx 1.68 \times 10^{11} \mathrm{\ W.h}$$
$$= 168 \mathrm{\ GW.h}$$

مشاهده می‌کنیم میزان مصرف انرژی برق روزانه آن تقریبا به اندازه مصرف برق روزانه کشور الجزایر می‌باشد.

سوال سوم

فرض می‌کنیم که X_t متغیر تصادفی تعداد رخدادهای پوآسون در یک بازه زمانی به طول t باشد و T نیز متغیر تصادفی زمان استخراج بلاک بعدی باشد. بنابراین داریم:

$$F_T(t) = P(T \leq t) = 1 - P(T > t) = 1 - P(X_t = 0)$$

$$= 1 - \frac{(\lambda t)^0}{0!} e^{-\lambda t} = 1 - e^{-\lambda t}$$

$$\Rightarrow f_T(t) = \frac{d}{dt} F_T(t) = \lambda e^{-\lambda t}$$

بنابراین همانگونه که مشاهده می‌شود نتیجه می‌گیریم که T دارای توزیع نمایی می‌باشد.

سوال چهارم

۱. هر نود از شبکه بلاکی را به عنوان پایان زنجیره‌ی بلوکی انتخاب می‌کند که آن را زودتر دریافت کند و با توجه به وجود تاخیر در شبکه هر کدام از نودها یکی از بلوک‌های A یا B را زودتر دریافت می‌کنند. در نهایت هر کدام از نودها شروع به ادامه دادن زنجیره بلوکی از بلوکی که زودتر دریافت کرده را می‌کند و بلوک دیگر را نیز پس از دریافت نگه خواهد داشت. حال هر کدام از این دو زنجیره که زودتر توسط نودهایی که بر روی آن کار می‌کنند، ادامه پیدا کند، به عنوان زنجیره اصلی جدید توسط همه پذیرفته می‌شود و بلاکی که زنجیره‌ی آن ادامه پیدا نکرده به اصطلاح یتیم می‌شود.

۲. اگر فرض کنیم نرخ استخراج بلاک برابر λ و تاخیر شبکه برابر Δ باشد، آنگاه به ازای هر بلاکی که به سر زنجیر اصلی اضافه می‌شود، به صورت میانگین $\lambda\Delta$ تا بلاک اضافی استخراج می‌شوند که یتیم خواهند شد. بنابراین می‌توان نرخ تولید بلاک‌های یتیم را به صورت زیر محاسبه کرد:

$$\frac{\lambda}{1+\Delta\lambda}\Delta\lambda = \frac{\Delta\lambda^2}{1+\Delta\lambda}$$

۳. در حال حاضر نرخ تولید بلوک‌های یتیم در حدود 1.8% می‌باشد.

۴. در بیت‌کوین اگر A نتواند بلوک تولید شده‌ی خود را وارد زنجیره اصلی کند، هیچ پاداشی دریافت نمی‌کند. از آن‌جا که بلوک B زودتر انتشار پیدا کرده بنابراین احتمال آنکه A بتواند وارد زنجیره اصلی شود بسیار کم است. در برخی دیگر از پروتوکل‌های و ارزهای دیجیتال پاداشی برای بلاک‌هایی که در زنجیر اصلی قرار نگرفته‌اند اما با اختلاف زمانی کمی از بلاک زنجیر اصلی استخراج شده‌اند نیز یک پاداش کمتر از پاداش اصلی در نظر گرفته می‌شود و به این بلاک‌ها uncle گفته می‌شود.

۵. نرخ یتیم شدن بلاک‌ها برای هر ماینر به چندین مورد بستگی دارد. اگر که یک ماینر قدرت پردازشی بیشتری داشته باشد، در نتیجه تعداد بیشتری بلاک پیدا خواهد کرد و به همین دلیل نرخ بلاک‌های یتیم بیشتری نیز خواهد داشت. همچنین اگر یک ماینر بتواند

بلاک‌های خود را سریع‌تر در شبکه ارسال کند، احتمال یتیم شدن بلاک‌های کمتر خواهد اما برای نودهایی که تاخیر زیادی در ارسال بلاک‌های خود دارند این نرخ بیشتر است.

سوال پنجم

از آن‌جا که سرویس‌دهنده به کلید خصوصی کاربر دسترسی ندارد، پس نمی‌تواند تراکنش جدیدی برای کاربر ایجاد کند که حمله double spend رخ دهد. تنها کاری که سرویس‌دهنده امکان آن را دارد، جلوگیری از ارسال پیام‌ها می‌باشد تا تراکنش‌های کاربر کامل نشود.

سوال ششم

Chebyshev

$$\Pr(X > \alpha n) \leq \frac{E(X)}{\alpha n} = \frac{np}{\alpha n} = \frac{p}{\alpha} = \frac{2}{5}$$

Markov

$$\Pr(X > \alpha n) = \Pr(|X - np| > (\alpha - p)n) \leq \frac{Var(X)}{(\alpha - p)^2 n^2}$$

$$= \frac{np(1-p)}{(\alpha - p)^2 n^2} = \frac{p(1-p)}{(\alpha - p)^2 n} = \frac{16}{n}$$

Chernoff

$$\Pr(X > \alpha n) \leq \min_s \frac{E(e^{sX})}{e^{s\alpha n}}$$

می‌توان X را به صورت حاصل جمع n متغیر تصادفی برنولی نوشت:

$$E(e^{sX}) = E\left(e^{s\sum_{i=0}^{n} Y_i}\right) = E(\Pi_{i=0}^{n} e^{sY_i}) = \left(E(e^{sY_i})\right)^n$$

$$= \left(pe^s + (1-p)\right)^n$$

$$\Rightarrow \frac{E(e^{sX})}{e^{s\alpha n}} \left(\frac{pe^s + (1-p)}{e^{s\alpha}}\right)^n$$

$$\frac{d}{ds}\left(\frac{pe^{s}+(1-p)}{e^{s\alpha}}\right)=(1-\alpha)pe^{(1-\alpha)s}-\alpha(1-p)e^{-\alpha s}=0$$

$$=e^{-\alpha s}\big((1-\alpha)pe^{s}-\alpha(1-p)\big)=0$$

$$\Rightarrow e^{s}=\frac{\alpha(1-p)}{(1-\alpha)p}=4\Rightarrow s=\ln 4$$

$$\Rightarrow \Pr(X>\alpha n)\le\left(0.4e^{-0.5}\right)^{n}$$

همان‌طور که مشاهده می‌شود به خصوص برای nهای بزرگ، Chernoff بسیار بهتر عمل می‌کند و کران پایین تری را می‌یابد.