

## سوال ۱

۱. به سوالات زیر پاسخ دهید:

- الف) مدل‌های صف با جمعیت نامحدود (infinite population) و جمعیت محدود (finite population) را توصیف کنید و تفاوت‌های آن‌ها را بیان نمایید.
- ب) انواع نظم‌دهی در صف‌ها یا نظام صف‌بندی (Queue discipline) چیست؟ سه مورد از این انواع را نام برده و هر کدام را به طور خلاصه شرح دهید.
- ج) جریان‌های شماره‌گذاری تصادفی (random-number streams) را توضیح دهید.

## جواب سوال ۱

الف) مدل‌های صف با جمعیت نامحدود و جمعیت محدود: جمعیت نامحدود به معنای آن است که تعداد مشتریان بالقوه برای ورود به صف بی‌نهایت است، بدین معنی که همیشه مشتری جدیدی برای ورود به صف وجود دارد و محدودیتی برای تعداد کل مشتریان وجود ندارد. این مدل معمولاً در سیستم‌هایی با ترافیک بالا مانند وب‌سایت‌ها یا سیستم‌های تلفنی مورد استفاده قرار می‌گیرد. در مقابل، جمعیت محدود به وضعیتی اشاره دارد که تعداد مشتریانی که ممکن است وارد صف شوند محدود است. این مدل در موقعیت‌هایی که تعداد کاربران سیستم محدود است، مثلاً در یک کسب‌وکار کوچک یا سیستمی با تعداد محدود کاربر، کاربرد دارد.

ب) انواع نظم‌دهی در صف‌ها:

- **FIFO (First In, First Out):** این رویه بر اساس ترتیب ورود مشتریان به صف عمل می‌کند. اولین مشتری که وارد صف می‌شود، اولین کسی است که خدمات دریافت می‌کند. این روش در اکثر موقعیت‌های رایج مانند صف‌های بانک یا فروشگاه‌ها به کار می‌رود.
- **LIFO (Last In, First Out):** در این روش، آخرین مشتری که وارد صف می‌شود، اولین کسی است که خدمات دریافت می‌کند. این روش عمدتاً در موقعیت‌های خاص مانند برخی فرآیندهای صنعتی یا در مدیریت داده‌ها به کار می‌رود.
- **اولویت‌بندی:** در این روش، مشتریان بر اساس اولویت‌های مختلف خدمات دریافت می‌کنند. اولویت‌ها می‌توانند بر اساس فاکتورهای مختلفی مانند اورژانسی بودن، وضعیت ویژه یا اهمیت مشتری تعیین شوند. این روش در موقعیت‌هایی مانند بیمارستان‌ها یا مراکز خدماتی که نیاز به سرویس‌دهی فوری دارند، کاربرد دارد.

ج) **جریان‌های شماره‌گذاری تصادفی:** جریان‌های شماره‌گذاری تصادفی ابزارهایی هستند که برای تولید دنباله‌هایی از اعداد تصادفی در شبیه‌سازی‌ها استفاده می‌شوند. این اعداد به منظور مدل‌سازی رفتارهای تصادفی در سیستم‌های واقعی استفاده می‌شوند، مانند تعیین زمان‌های ورود تصادفی مشتریان به صف یا زمان‌های متفاوت سرویس‌دهی در یک مرکز خدماتی. استفاده از این جریان‌ها امکان پذیرش یک مدل شبیه‌سازی را که به‌خوبی پویایی‌های واقعی را بازتاب می‌دهد، فراهم می‌کند.

## سوال ۲

سوال: اعداد تصادفی زیر را در نظر بگیرید:

۰/۰۵, ۰/۰۸, ۰/۱۴, ۰/۲۴, ۰/۳۳, ۰/۳۳, ۰/۳۹,

۰/۴۱, ۰/۴۴, ۰/۵۳, ۰/۵۶, ۰/۵۸, ۰/۶۳, ۰/۷۳,

۰/۷۶, ۰/۸۳, ۰/۸۴, ۰/۸۸, ۰/۸۸, ۰/۹۳

فرض اینکه این اعداد توزیع یکنواخت داشته باشند را با اعمال تست **Kolmogorov-Smirnov** در بخش الف (با سطح معناداری ۵٪) بررسی کنید. و در نظر گرفتن ۱۰ بازه برای اعداد تصادفی تولید شده، تست بخش قبل را این بار با روش **Chi-Square** در بخش ب (با سطح معناداری ۵٪) بررسی کنید. در بخش ج، با توضیح دلیل تعیین کنید که از نتیجه کدام تست باید استفاده کرد.

## جواب سوال ۲

طبق کد پیوست شده برای این سوال، داریم که:

**الف) تست: Kolmogorov-Smirnov**

آماره تست: ۰/۱۳

مقدار  $p$ : ۰/۸۴۶

با توجه به مقدار  $p$  بزرگتر از ۰/۰۵، فرض توزیع یکنواخت رد نمی شود.

**ب) تست: Chi-Square**

آماره تست: ۵/۰

مقدار  $p$ : ۰/۸۳۴

با توجه به مقدار  $p$  بزرگتر از ۰/۰۵، فرض توزیع یکنواخت رد نمی شود.

**ج) انتخاب تست مناسب:**

با توجه به نتایج مشابه در هر دو تست، هر دو تست نشان می دهند که فرض توزیع یکنواخت برای این داده ها قابل قبول است. با این حال، انتخاب تست مناسب بستگی به شرایط خاص مطالعه و نوع داده ها دارد. تست **Kolmogorov-Smirnov** برای مجموعه داده های کوچکتر مناسب است، در حالی که تست **Chi-Square** برای مجموعه داده های بزرگتر با تقسیم بندی به بازه ها مفید است.

### سوال ۳

در این سوال قصد بررسی مدل صف در یک مطب را داریم. فرض کنید که در یک مطب دو پزشک وجود دارند که هرکدام از آنها در ۱۵ دقیقه، یک بیمار را معاینه و درمان می‌کنند و در هر یک ساعت، یک بیمار به مطب وارد می‌شود. فرض کنید که میدانیم این مطب امروز تنها ۱۰ بیمار خواهد داشت.

الف) تعداد میانگین افرادی که در مطب حضور دارند را بدست آورید.

ب) هر بیمار به طور میانگین چند دقیقه را در صف می‌گذارند؟

ج) هر پزشک به طور میانگین در چه نسبتی از یک ساعت، هیچ بیماری را ویزیت نمی‌کند؟

### جواب سوال ۳

میانگین تعداد افراد در مطب: از آنجایی که هر پزشک در هر ۱۵ دقیقه یک بیمار را معاینه می‌کند و هر ساعت یک بیمار وارد می‌شود، هر پزشک می‌تواند در هر ساعت ۴ بیمار را ویزیت کند. با دو پزشک موجود، آن‌ها می‌توانند در مجموع ۸ بیمار را در هر ساعت ویزیت کنند. با این حال، تنها یک بیمار در هر ساعت وارد می‌شود، بنابراین، در هر ساعت به طور میانگین  $\frac{1}{8}$  بیمار در مطب حضور خواهد داشت.

با استفاده از مدل صف  $M/M/c/\infty/\infty$ ، محاسبات زیر انجام شده‌اند:

- میانگین تعداد افراد در مطب: حدود ۰/۲۵۴ نفر.
- میانگین زمان انتظار هر بیمار در صف: حدود ۱۵/۲۴ دقیقه.
- میانگین زمان بیکاری هر پزشک: حدود ۸۷/۵ درصد.

### سوال ۴

سوال:

فرض کنید یک سیستم صف  $M/M/1$  وجود دارد.

الف) اثبات کنید که احتمال وجود  $n$  مشتری در مغازه برابر است با:

$$P_n = (1 - \rho)\rho^n$$

که در آن  $\rho = \frac{\lambda}{\mu}$  نرخ استفاده (utilization) است.

ب) با استفاده از نتیجه بخش الف، اثبات کنید که تعداد مشتریان منتظر در صف به طور میانگین برابر است با:

$$L_q = \frac{\rho^2}{1 - \rho}$$

## جواب سوال ۴

الف) برای یک سیستم صف  $M/M/1$ ، فرمول استاندارد برای محاسبه احتمال وجود  $n$  مشتری در سیستم (هم در صف و هم در خدمت) استفاده می‌شود. این فرمول بر اساس این واقعیت است که احتمال ورود مشتریان به سیستم به صورت هندسی توزیع شده است. فرض کنید  $\rho = \frac{\lambda}{\mu}$  که  $\lambda$  نرخ ورود و  $\mu$  نرخ خدمت است. فرض می‌کنیم که سیستم در حالت تعادل است. پس داریم:

$$P_n = (1 - \rho)\rho^n$$

که در آن  $P_n$  احتمال وجود دقیقاً  $n$  مشتری در سیستم است. این فرمول از معادلات تعادل حالت پایا برای سیستم‌های صف  $M/M/1$  به دست می‌آید.

ب) برای محاسبه تعداد میانگین مشتریان منتظر در صف، از فرمول Little استفاده می‌کنیم. فرمول Little می‌گوید که  $L = \lambda W$ ، که  $L$  تعداد میانگین مشتریان در سیستم و  $W$  میانگین زمان انتظار یک مشتری در سیستم است. برای یک سیستم  $M/M/1$ ، میانگین زمان انتظار در سیستم  $W$  برابر است با  $\frac{1}{\mu - \lambda}$ . پس داریم:

$$L = \lambda \times \frac{1}{\mu - \lambda} = \frac{\lambda}{\mu - \lambda} = \frac{\rho}{1 - \rho}$$

از آنجا که  $L_q = L - \rho$  (که  $L_q$  تعداد میانگین مشتریان در صف است)، می‌توانیم بنویسیم:

$$L_q = \frac{\rho}{1 - \rho} - \rho = \frac{\rho^2}{1 - \rho}$$

که این فرمول نشان دهنده تعداد میانگین مشتریان منتظر در صف در یک سیستم صف  $M/M/1$  است.

## سوال ۵

پس از راه‌اندازی آزادراه تهران-شمال، وزارت راه و شهرسازی قصد دارد بین استفاده از باجه‌ی اتوماتیک و یا باجه‌ی سنتی برای دریافت عوارضی یکی را انتخاب کند. هزینه‌ی باجه‌ی اتوماتیک دو برابر باجه‌های سنتی است و بودجه این وزارت‌خانه برای این پروژه برابر هزینه‌ی یک باجه‌ی اتوماتیک است. بنابراین می‌تواند یا یک باجه‌ی اتوماتیک و یا دو باجه‌ی سنتی قرار دهد. همچنین فرض کنید خودروها را با یک فرآیند پواسون با نرخ ۱۰۰۰ نفر در ساعت مدل کنیم که به عوارضی یک شهر می‌رسند و در یک صف طولانی قرار می‌گیرند. با توجه به زمان انتظار مورد انتظار، کدام انتخاب برای وزارت راه و شهرسازی بهتر است؟

الف) این صف توسط یک باجه‌ی خودکار با نرخ ۱۲۰۰ نفر در ساعت خدمت رسانی می‌شود. همچنین فرض کنید که زمان‌های سرویس‌دهی از توزیع نمایی پیروی کنند و استراتژی سرویس‌دهی FIFO است. پیدا کنید زمان انتظار مورد انتظار در حالت پایدار (Steady-state expected waiting time).

ب) صف توسط دو باجه‌ی سنتی با نرخ سرویس‌دهی ۶۰۰ نفر در ساعت سرویس‌دهی شود. مشابه قسمت قبل فرض کنید که زمان‌های سرویس‌دهی از توزیع نمایی پیروی کنند و استراتژی سرویس‌دهی FIFO است و پیدا کنید زمان انتظار مورد انتظار در حالت پایدار (Steady-state expected waiting time).

## جواب سوال ۵

برای محاسبه زمان انتظار مورد انتظار در هر دو سناریو، از فرمول‌های مربوط به سیستم‌های صف  $M/M/1$  و  $M/M/2$  استفاده می‌کنیم.

الف) برای یک باجه اتوماتیک با نرخ خدمت دهی ۱۲۰۰ نفر در ساعت، نرخ استفاده  $\rho$  برابر است با  $\frac{1}{12}$ . زمان انتظار مورد انتظار در حالت پایدار به صورت زیر محاسبه می‌شود:

$$W = \frac{1}{\mu - \lambda} = \frac{1}{1200 - 1000} = 0.005 \text{ ساعت (یا ۱۸ ثانیه)}$$

ب) برای دو باجه سنتی با نرخ خدمت دهی ۶۰۰ نفر در ساعت برای هر باجه، نرخ استفاده  $\rho$  برابر است با  $\frac{1}{2 \times 6}$ . زمان انتظار مورد انتظار در حالت پایدار به صورت زیر محاسبه می‌شود:

$$W = \frac{1}{2 \times \mu - \lambda} = \frac{1}{2 \times 600 - 1000} = 0.005 \text{ ساعت (یا ۱۸ ثانیه)}$$

بنابراین، از نظر زمان انتظار، هیچ تفاوت قابل توجهی بین این دو گزینه وجود ندارد. انتخاب بین این دو گزینه باید بر اساس عوامل دیگری مانند هزینه، راحتی استفاده و غیره انجام شود.

## سوال ۶

در این بخش قصد پیاده سازی الگوریتم های تولید اعداد شبه تصادفی و روش های تست آنها را داریم.

**بخش اول: پیاده سازی الگوریتم های تولید اعداد شبه تصادفی:**

- در ابتدا یک تابع تولید عدد با الگوریتم CLCG را پیاده کنید و بعد با استفاده از این تابع، الگوریتم LCM را به صورت یک تابع پیاده کنید. در انتخاب پارامترهای الگوریتم ها آزاد هستید.
- در آخر، به عنوان یک روش متفاوت دیگر تولید اعداد شبه تصادفی، الگوریتم xor-shift را مشابه بخش قبلی، پیاده کنید. (همچنین در مستند تمرین این بخش، توضیحی در مورد این الگوریتم ارائه دهید.)

**بخش دوم: تولید اعداد**

- به ازای هرکدام از روش های پیاده سازی شده در بخش قبل، ۱۰۰۰ عدد ایجاد کنید و نمودار هیستوگرام هرکدام از آنها را رسم کنید.

**بخش سوم: تست اعداد**

- در این بخش باید سه روش KS-test، chi-square test و auto-correlation test را پیاده کنید.
- KS-test: این تست را به صورت یک تابع ایجاد کنید که با ورودی گرفتن اعداد تولید شده و مقدار level of significance نتیجه تست را چاپ کند.
- Chi-square test: این تست نیز مشابه بخش قبل پیاده شود.

– Auto-correlation test: این تست در قالب یک تابع ایجاد شود که با ورودی گرفتن اعداد تولید شده به همراه level of significance و اندیس اولین عدد در تست و مقدار lag، نتیجه تست را ایجاد کند.

- برای این بخش تمرین می توانید از یکی از زبان های برنامه نویسی پایتون، جاوا، سی یا سی پلاس پلاس استفاده کنید. برای ارسال پاسخ این بخش، به همراه کد پیاده سازی باید یک مستند حاوی توضیحات لازم برای بخش های مختلف کد را ارائه دهید.
- نکته: برای بدست آوردن مقادیر بحرانی یا p-value در تست های مختلف، میتوانید از کتابخانه های در دسترس در زبان برنامه نویسی مد نظر استفاده کنید و در پیاده سازی شما این مقادیر نباید hard code شوند.

## جواب سوال ۶

### توضیح الگوریتم xor-shift :

الگوریتم xor-shift یکی از روش های تولید کننده اعداد شبه تصادفی است که بر پایه عملیات های بیتی پایه ای نظیر XOR (تفاوت منطقی) و Shift (جابجایی بیتی) عمل می کند. این الگوریتم از یک عدد اولیه (seed) شروع کرده و با انجام یک سری عملیات های xor و shift بر روی این عدد، عدد جدیدی را تولید می کند. فرایند کاری الگوریتم به این صورت است:

- ابتدا عدد اولیه (seed) با استفاده از عملیات shift به چپ و سپس عملیات xor با خودش ترکیب می شود.
- سپس عدد حاصل از مرحله قبلی با استفاده از عملیات shift به راست و دوباره عملیات xor با خودش ترکیب می شود.
- در آخر، عدد حاصل از مرحله دوم دوباره با استفاده از عملیات shift به چپ و عملیات xor با خودش ترکیب می شود تا عدد شبه تصادفی جدیدی به دست آید.

این الگوریتم به دلیل سادگی و کارایی بالا در تولید اعداد شبه تصادفی محبوب است و در بسیاری از کاربردها استفاده می شود.