



دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)
دانشکده مهندسی صنایع و سیستم‌های مدیریت

پروژه درس اصول شبیه‌سازی

اعضا:

حامد اعراب - ۹۹۲۵۰۰۳

شهریار خلوتی - ۹۹۲۵۰۱۵

استاد: دکتر عباس احمدی

تدریس‌یار: مهدی محمدی

بهار ۱۴۰۳

فهرست

۱	شرح مسئله
۲	مقدمه
۲	چارچوبی برای شبیه‌سازی رویدادهای گسسته
۳	کنترلر
۵	رویداد
۷	مدل‌سازی و حل مسئله‌ها
۷	موجودیت‌ها
۱۱	رویدادها
۱۴	نتایج
۱۴	راه‌اندازی
۱۵	اجرای اصلی
۱۶	توزیع‌ها
۲۱	تعداد بهینه پزشکان
۲۲	نتیجه‌گیری

شرح مسئله

به اورژانس یک بیمارستان هر 4 ± 20 دقیقه یک بیمار وارد می‌شود. اورژانس دارای دو پزشک است. ۱۵ درصد بیماران کسانی هستند که نیاز به رسیدگی فوری دارند اما بقیه می‌توانند صبر کنند. به بیماران دسته اول بالاترین درجه اولویت، یعنی اولویت ۳ داده می‌شود تا در اسرع وقت، پزشکی را به مدت 5 ± 30 دقیقه ببینند. ولی بعد اولویت آنها به ۲ کاهش می‌یابد و به انتظار می‌مانند تا دوباره یک پزشک آزاد شود تا این بار به مدت 13 ± 28 دقیقه مداوا و سپس مرخص شوند. بیماران دسته دوم ابتدا اولویت ۱ می‌گیرند و هنگامی که نوبتشان برسد به مدت 10 ± 15 دقیقه درمان می‌شوند، سپس اولویتشان به ۲ افزایش می‌یابد. به انتظار می‌مانند تا یک پزشک آزاد شود و به مدت 7 ± 10 دقیقه، آنها را به طور نهایی درمان کند و سرانجام مرخص شوند.

الف) ابتدا نمودار کنترلر شبیه‌سازی و نمودارهای جریان را برای پیشامدهای اصلی ترسیم نمایید. سپس مدل کامپیوتری شبیه‌سازی را برای ۳۰ روز ۲۴ ساعته اجرا نمایید. به منظور ایجاد باری از بیماران در سیستم، پیش از اجرای ۳۰ روزه فوق، یک دوره راه‌اندازی ۲ روزه در نظر بگیرید. شرایط را برای دوره ۲ روز راه‌اندازی و دوره ۳۰ روزه به‌طور جداگانه گزارش کنید. در صورت امکان، از برآوردهای نقطه‌ای و فاصله‌ای استفاده نمایید.

ب) متوسط و ماکزیمم طول صف را برای بیماران (دسته اول و دوم) به تفکیک صف‌ها محاسبه کنید. چند درصد از بیماران مجبور به انتظار در صف نیستند؟ نمودار فراوانی مدت انتظار در صف را برای بیماران به تفکیک صف ترسیم کنید. چند درصد از بیماران برای دیدن اولین پزشک کمتر از ۴ دقیقه صبر می‌کنند؟ مقادیر زمان انتظار برای بیماران را به تفکیک دو دسته به همراه نمودار تابع توزیع ارائه نمایید. درصد اشتغال پزشکان را محاسبه کنید.

ج) تعداد بهینه پزشک‌ها را با هدف حداقل کردن کل زمان انتظار و ماکزیمم کردن درصد اشتغال پزشکان بیابید.

مقدمه

در این مسئله، سیستمی داریم که شامل دو نوع سرور است: پزشک معاینه‌گر و پزشک درمانگر. مراجعه‌کنندگان برای دسترسی به پزشک معاینه‌گر باید در یکی از دو صف با اولویت بالا و پایین قرار گیرند، در حالی که برای پزشک درمانگر تنها یک صف وجود دارد. رویدادهای سیستم به ورود و خروج مراجعه‌کنندگان محدود می‌شود و سرورها به صورت تصادفی انتخاب می‌شوند.

هدف این شبیه‌سازی، ارزیابی عملکرد سیستم از طریق تحلیل معیارهای مختلف است. معیارهایی نظیر حداکثر طول هر صف و میانگین طول صف‌ها بررسی می‌شوند. همچنین، میانگین درصد اشتغال سرورها و میانگین درصد انتظار بیماران محاسبه می‌شود. به علاوه، درصد بیمارانی که بدون معطلی خدمات دریافت می‌کنند و درصد بیمارانی که زمان انتظار کمی (زیر ۴ دقیقه) دارند نیز ارزیابی می‌گردد.

با تحلیل این معیارها، می‌توان بهترین تعداد سرورها را برای هر نوع مشخص کرد تا هم میانگین زمان انتظار بیماران به حداقل برسد و هم درصد زمان اشتغال سرورها به حداکثر برسد. این ارزیابی‌ها به بهبود کیفیت خدمات و کارایی سیستم کمک می‌کند.

چارچوبی برای شبیه‌سازی رویدادهای گسسته

از ابتدای این ترم، تصمیم گرفتیم که یک چارچوب^۱ و بنیان واحد برای حل مسائل شبیه‌سازی طراحی و پیاده کنیم. این چارچوب باید ویژگی‌های زیر را داشته باشد:

^۱ Framework

(۱) گسترش‌پذیری^۲: در مسائل گوناگون با سیستم‌هایی سروکار داریم که متغیرها^۳، وضعیت^۴، منطق^۵، و رفتار^۶ مخصوص خود را دارند. بنابراین، به چارچوبی نیاز داریم که حداقل امکانات شبیه‌سازی را فراهم کرده و اجازه‌ی افزودن سایر اجزای مورد نیاز را بدهد.

(۲) مقیاس‌پذیری^۷: مسائل شبیه‌سازی در پیچیدگی^۸ و اندازه با هم تفاوت دارند. مثلاً ممکن است تعداد صف‌ها، انواع و مراحل خدمت‌دهی، و تعداد خدمت‌دهنده‌ها در هر مسئله متفاوت باشند. بنابراین، چارچوب مورد استفاده باید قابلیت به‌کارگیری را در مقیاس‌های مختلف داشته باشد.

(۳) بهره‌وری^۹: چارچوب ما باید در اجرای فرآیند شبیه‌سازی سریع بوده و مصرف منابع کمی داشته باشد. این منابع شامل انرژی، حافظه، و فضای ذخیره‌سازی‌اند. با داشتن بهره‌وری مناسب می‌توان یک سیستم را به دفعات بیش‌تری شبیه‌سازی، و از صحت خروجی‌ها و درستی مدل شبیه‌سازی‌مان اطمینان حاصل کرد.

با توجه به این ویژگی‌ها، تصمیم بر آن شد که چارچوب ما دو موجودیت کنترلر^{۱۰} و رویداد را در سطحی پایه‌ای و بنیادین برای استفاده فراهم کند. بدین ترتیب می‌توانیم این دو موجودیت را متناسب با شرایط هر مسئله گسترش داده و سیستم را مدل‌سازی کنیم. در ادامه، به جزئیات آن‌ها بیش‌تر می‌پردازیم.

کنترلر

کنترلر وظیفه‌ی اجرا و مدیریت فرآیند شبیه‌سازی را بر عهده دارد. به زبان ساده، با شروع فرآیند شبیه‌سازی، رویدادها به کنترلر ارسال^{۱۱} می‌شوند. کنترلر آن‌ها را به «فهرست رویدادهای آینده^{۱۲}» می‌افزاید و این فهرست را بر اساس «زمان مقرر^{۱۳}» رویدادها مرتب می‌کند. پس از پایان شلیک^{۱۴} رویداد پیشین، رویداد پیش رو^{۱۵} از FEL خارج و پس از جلو

² Extensibility

³ Variables

⁴ State

⁵ Logic

⁶ Behavior

⁷ Scalability

⁸ Complexity

⁹ Efficiency

¹⁰ Controller

¹¹ Dispatch

¹² Future Events List (FEL)

¹³ Due Time

¹⁴ Triggering

¹⁵ Upcoming Event

بردن ساعت شبیه‌سازی^{۱۶} شلیک می‌گردد. در این میان، اگر ساعت شبیه‌سازی به زمان مشخص شده برای پایان برسد یا از آن فراتر رود، شبیه‌سازی متوقف می‌گردد.

کنترلر دو پارامتر رویداد نخستین^{۱۷} و زمان پایان شبیه‌سازی^{۱۸} را دریافت و صحت‌سنجی^{۱۹} می‌کند. سنجش صحت پارامترها عبارتند از: (۱) مثبت بودن زمان پایان شبیه‌سازی و (۲) صفر بودن فاصله زمانی رویداد نخستین. پس از سنجش صحت پارامترها، رویداد نخستین به FEL افزوده شده و ساعت شبیه‌سازی برابر با دقیقه صفر قرار داده می‌شود. لازم به ذکر است که در این چارچوب، تمامی زمان‌ها با واحد دقیقه به‌کار می‌روند. خلاصه‌ای از پارامترها و متغیرهای مربوط به کنترلر را می‌توانید مشاهده کنید:

پارامترها		متغیرها	
نام	توضیحات	نام	توضیحات
initialEvent	رویداد نخستین	futureEvents	فهرست رویدادهای آینده
stopTime	زمان پایان شبیه‌سازی	clock	ساعت شبیه‌سازی

حال، کنترلر برای شروع فرآیند شبیه‌سازی آماده است. با شروع فرآیند، رویداد نخستین که به FEL افزوده شده بود را از این فهرست خارج و سپس شلیک می‌کنیم. در پی آن، رویدادهای بعدی دریافت شده و به FEL افزوده می‌گردند و این حلقه^{۲۰} تکرار می‌گردد، تا این که به زمان پایان شبیه‌سازی برسیم. در شکل بعدی، نمودار جریان را برای فرآیند شبیه‌سازی کنترلر می‌توانید مشاهده کنید. این نمودار همان توضیحات ساده‌ای که پیش‌تر داده شدند را با جزئیات بیشتر ارائه می‌دهد.

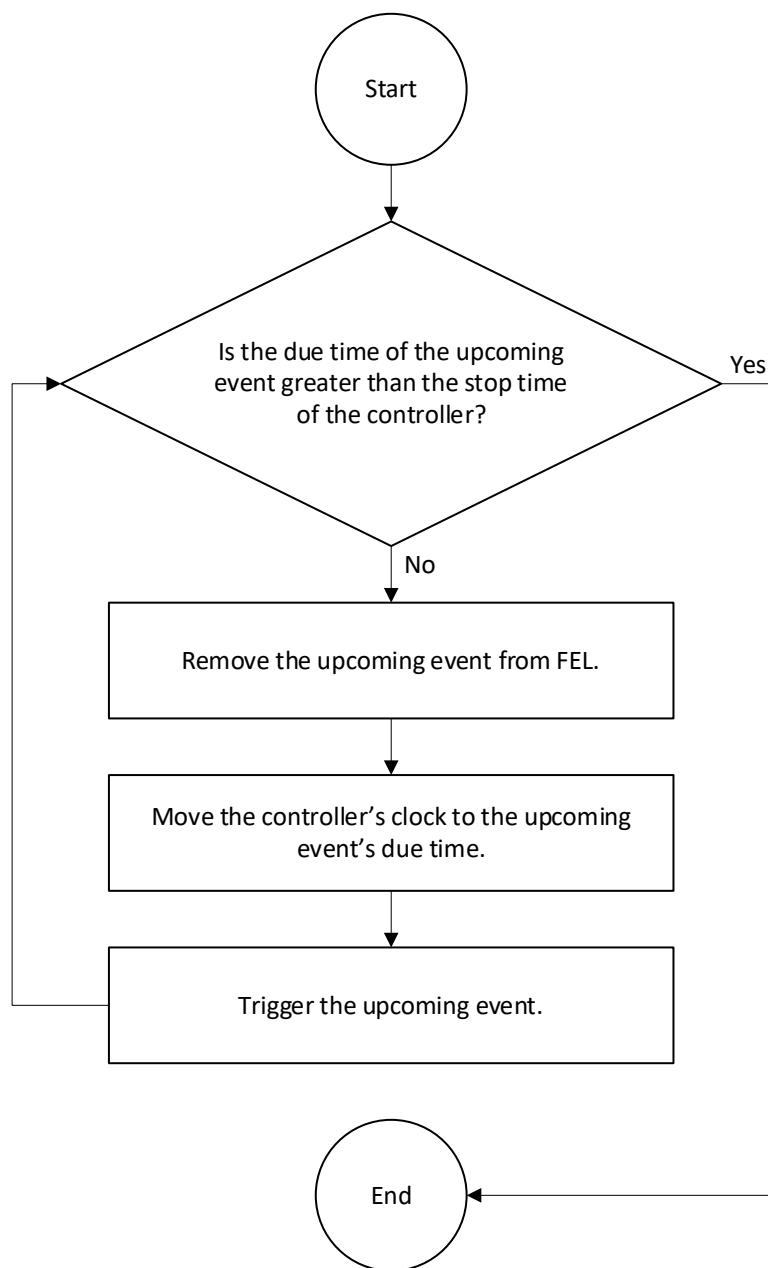
¹⁶ Simulation Clock

¹⁷ Initial Event

¹⁸ Simulation Stop Time

¹⁹ Validate

²⁰ Loop



رویداد

این موجودیت بستر را برای ایجاد رویدادهای مختلف در شبیه‌سازی یک سیستم فراهم می‌کند. تنها پارامتری که دریافت می‌کند فاصله^{۲۱} زمان شلیک آن با ساعت شبیه‌سازی است و پس از آن که به کنترلر ارسال شد، «زمان مقرر» آن برای شلیک مشخص می‌گردد. در نهایت، کنترلر رویداد را به FEL افزوده و فهرست بر اساس زمان مقرر مرتب می‌کند.

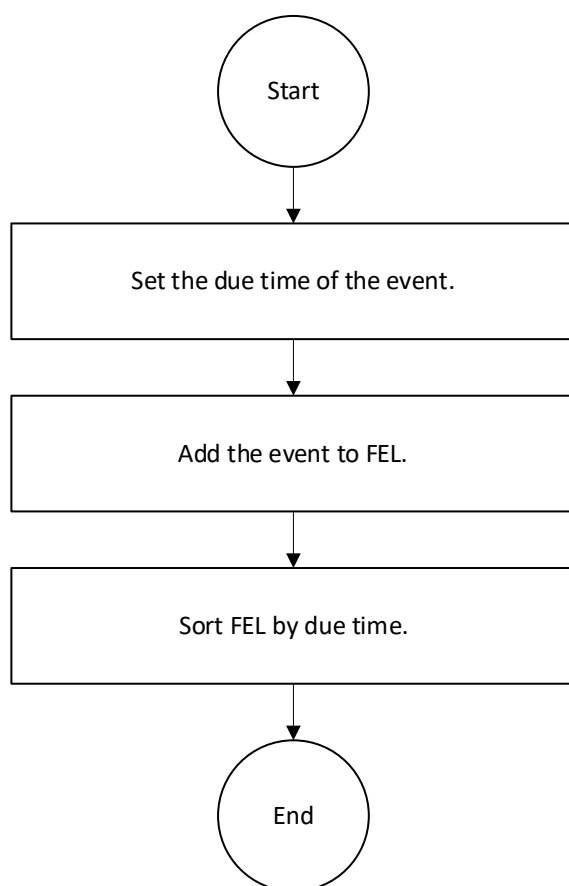
²¹ Interval

$$dueTime_{event} = clock + interval_{event}$$

خلاصه‌ای از پارامترها و متغیرهای مربوط به رویداد را می‌توانید مشاهده کنید:

متغیرها		پارامترها	
توضیحات	نام	توضیحات	نام
زمان مقرر	dueTime	فاصله زمانی	interval

در ادامه، نمودار جریان را برای ارسال یک رویداد به کنترلر شبیه‌سازی مشاهده می‌کنید.



در ادامه به نحوه مدل‌سازی و رسیدن به جواب هر مسئله با استفاده از این چارچوب می‌پردازیم.

مدل سازی و حل مسئله‌ها

برای حل مسئله با استفاده از چارچوب پیشنهادی، کنترلری طراحی می‌شود که رفتارهای بنیادین و پایه‌ای را از کنترلر چارچوب به ارث می‌برد. به نوعی، ما به این مسائل با دیدی شیء‌گرا^{۲۲} نگاه کرده و در کنار کنترلر، رویدادهای شبیه‌سازی را نیز با ارث‌بری^{۲۳} از رویداد بنیادین پیاده می‌کنیم.

موجودیت‌ها

بیمار

در این سیستم، بیمار همان مشتری است که خدمت دریافت می‌کند؛ از این رو، پارامترهای زیر به تفکیک مرحله (درمان/معاینه) برایش تعریف می‌شود:

- زمان ورود
- زمان خدمت
- زمان خروج
- پزشک مربوطه
- زمان انتظار
- اولویت (کم/زیاد)

پارامتر زمان انتظار در هر مرحله از رابطه‌ی زیر به دست می‌آید:

$$t_{\text{waiting}} = t_{\text{departure}} - t_{\text{service duration}} - t_{\text{arrival}}$$

و مقدار مجموع زمان انتظار از جمع زمان‌های انتظار هر مرحله:

$$t_{\text{waiting, total}} = t_{\text{waiting, examination stage}} + t_{\text{waiting, treatment stage}}$$

²² Object-Oriented

²³ Inheritance

پارامتر زمان کل نیز از رابطه‌ی زیر به دست می‌آید:

$$t_{\text{duration}} = t_{\text{departure, treatment}} - t_{\text{arrival, examination}}$$

هر بیمار که وارد بیمارستان می‌شود، یا وضعیت وخیم و اورژانسی دارد یا وضعی قابل کنترل. اگر وضعیتش اورژانسی باشد، برای مرحله اول (معاینه) اولویت ۳ که بیشترین اولویت است را می‌گیرد و سپس به اولویت ۲ کاهش پیدا می‌کند. اگر وضعیتش غیراورژانسی باشد، برای مرحله اول اولویت ۱ می‌گیرد و سپس به اولویت ۲ افزایش پیدا می‌کند. به همین دلیل، پارامتری با نام اولویت نیز تعریف می‌شود که نوع بیمار را مشخص کند. از آنجا که در فاز نتیجه‌گیری، می‌خواهیم آماره‌های متفاوتی برای هر دو نوع اورژانسی و غیراورژانسی داشته باشیم، موجودیت بیمار به دو موجودیت بیمار اورژانسی^{۲۴} و غیر اورژانسی^{۲۵} تقسیم می‌شود که البته تمام پارامترهای موجودیت بیمار را به ارث می‌برند.

پزشک

همانطور که بیمار به مثابه‌ی مشتری سیستم است، پزشک نیز سرور یا خدمت‌دهنده است و پارامترهای زیر برایش تعریف می‌شود:

- وضعیت (آزاد/مشغول)
- زمان اشتغال
- بیماران درمان شده
- مرحله (معاینه/درمان)

مقدار زمان اشتغال با محاسبه‌ی مجموع زمان‌های درمان بیماران درمان‌شده توسط همان پزشک در همان مرحله به دست می‌آید.

با توجه به اینکه که در این سیستم دو مرحله‌ی معاینه و درمان وجود دارد و آماره‌های متفاوتی برای هر بخش می‌خواهیم، دو موجودیت پزشک معاینه‌گر و پزشک درمانگر تعریف می‌شود که تمامی پارامترهای موجودیت پزشک را به ارث می‌برند و تنها تفاوتشان در نوع کارشان است.

²⁴ High Priority

²⁵ Low Priority

کنترلر

کنترلر این سیستم فرآیند شبیه‌سازی را از چارچوب به‌ارث می‌برد. اما نیاز است تا پارامترها، متغیرها، و خروجی‌های شبیه‌سازی مختص به این مسئله را به آن اضافه کنیم. رویکرد انتخاب سرور به صورت رندوم است و از بین پزشکان آزاد، یک نفر به صورت رندوم به عنوان خدمت‌دهنده به بیمار بعدی انتخاب می‌شود. پارامترهای دیگر کنترلر، تعداد دکترهای معاینه‌گر و تعداد دکترهای درمانگر است. در مسئله‌ی اصلی، مقدار هر دو پارامتر برابر ۱ است، اما هنگام محاسبه‌ی تعداد دکتر بهینه، این دو پارامتر مقادیر متفاوتی می‌گیرند.

سه صف سیستم، صف بیماران اورژانسی، صف بیماران عادی، و صف مرحله‌ی درمان است که مورد آخر بین هر دو نوع بیمار مشترک است. هر صف، با ورود بیمار جدید، میانگین طول و مقدار بیشینه طول را در صورت تغییر به‌روز می‌کند.

موارد دیگری که در کنترلر تعریف می‌شوند:

- مدت زمان گرم کردن
- بیماران درمان‌شده
- پزشکان درمانگر
- پزشکان معاینه‌گر

خروجی‌هایی که از سیستم انتظار داریم و مقادیرشان محاسبه می‌شود:

- حداکثر طول هر صف
- میانگین طول هر صف
- میانگین درصد اشتغال
- تعداد کل بیماران

$$\text{average utilization percentage} = \frac{\text{total utilization time of all doctors}}{\text{number of doctors} * \text{total time}}$$

- میانگین درصد انتظار

$$\text{average waiting percentage} = \frac{\text{total waiting time of all patients}}{\text{total duration of all patients}}$$

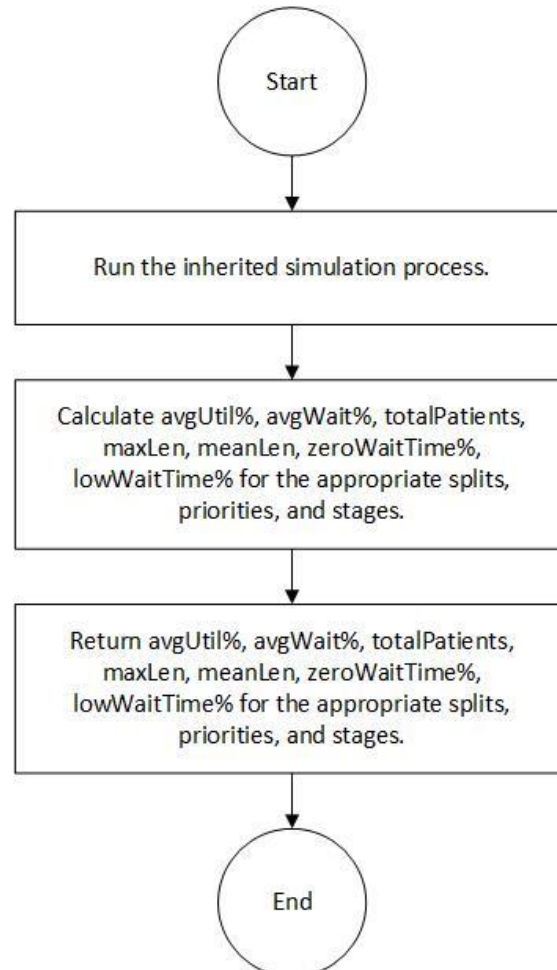
- درصد بیماران بدون معطلی برای هر صف

$$\text{zero wait time percentage} = \frac{\text{number of patients with 0 waiting}}{\text{total number of patients}}$$

- درصد بیماران کم معطلی (زیر ۴ دقیقه) برای هر صف

$$\text{low wait time percentage} = \frac{\text{number of patients with waiting} < 4}{\text{total number of patients}}$$

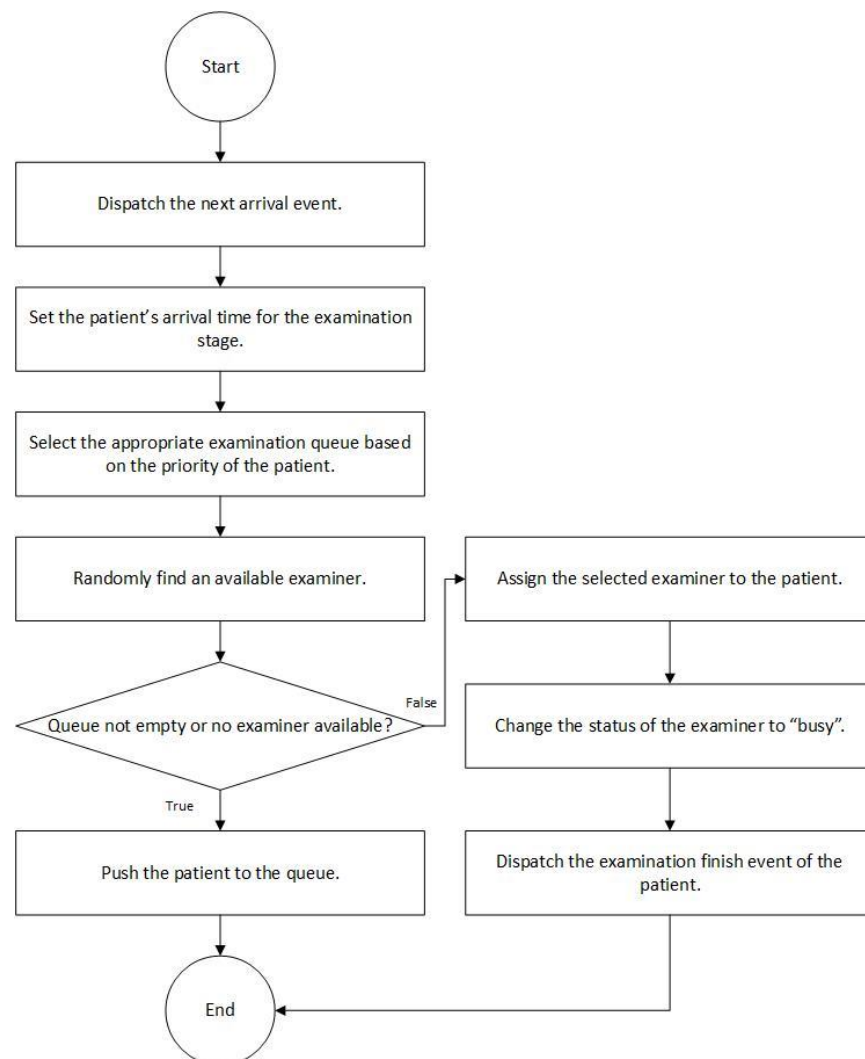
در نهایت، در شکل زیر، نمودار جریان فرآیند شبیه‌سازی کنترلر شبیه‌سازی سیستم را ببینید:



رویدادها

ورود

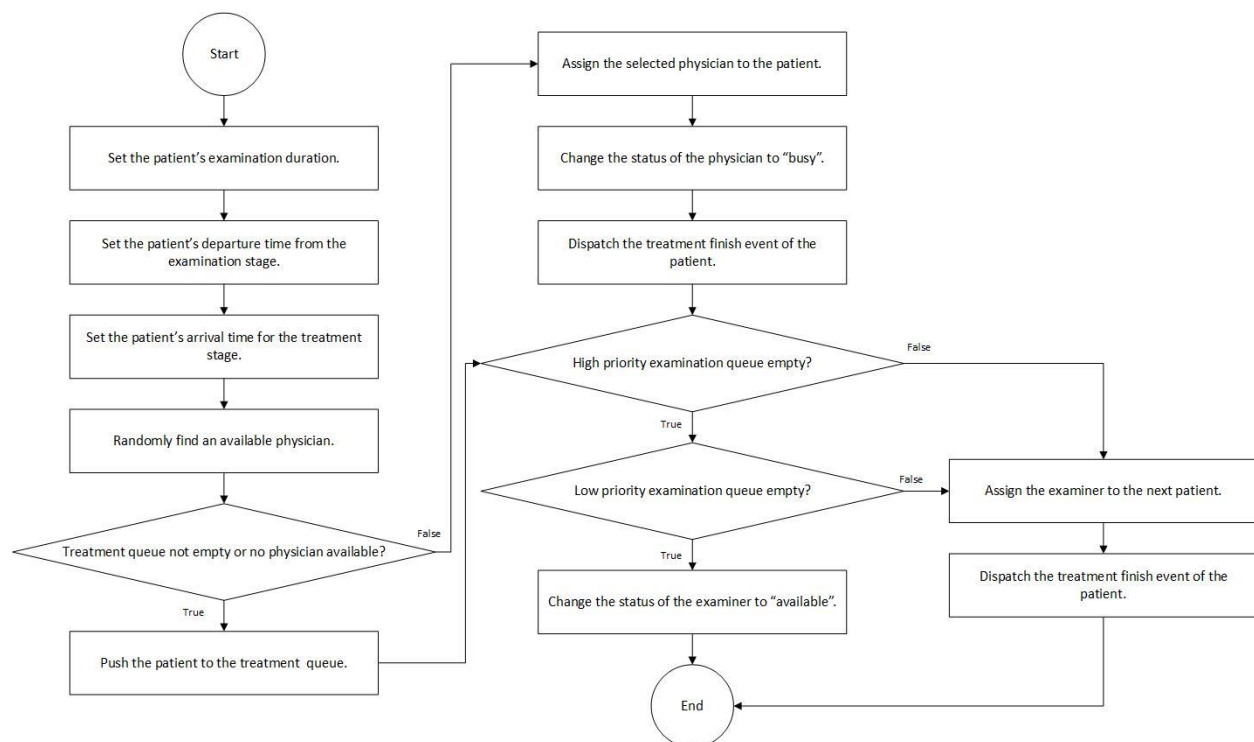
فرایند با شروع رویداد ورود بیمار آغاز می‌شود. ابتدا سیستم زمان ورود بیمار به مرحله معاینه را تنظیم می‌کند. سپس بیمار بر اساس اولویت به صف معاینه مناسب اختصاص داده می‌شود. در مرحله بعد، سیستم به صورت تصادفی یک معاینه‌گر موجود را پیدا می‌کند. اگر صف خالی نباشد یا معاینه‌گری موجود نباشد، بیمار به صف اضافه می‌شود. در غیر این صورت، معاینه‌گر انتخاب شده به بیمار اختصاص داده شده و وضعیت معاینه‌گر به "مشغول" تغییر می‌کند. سپس رویداد پایان معاینه بیمار اعزام می‌شود و فرایند به پایان می‌رسد.



اتمام معاینه

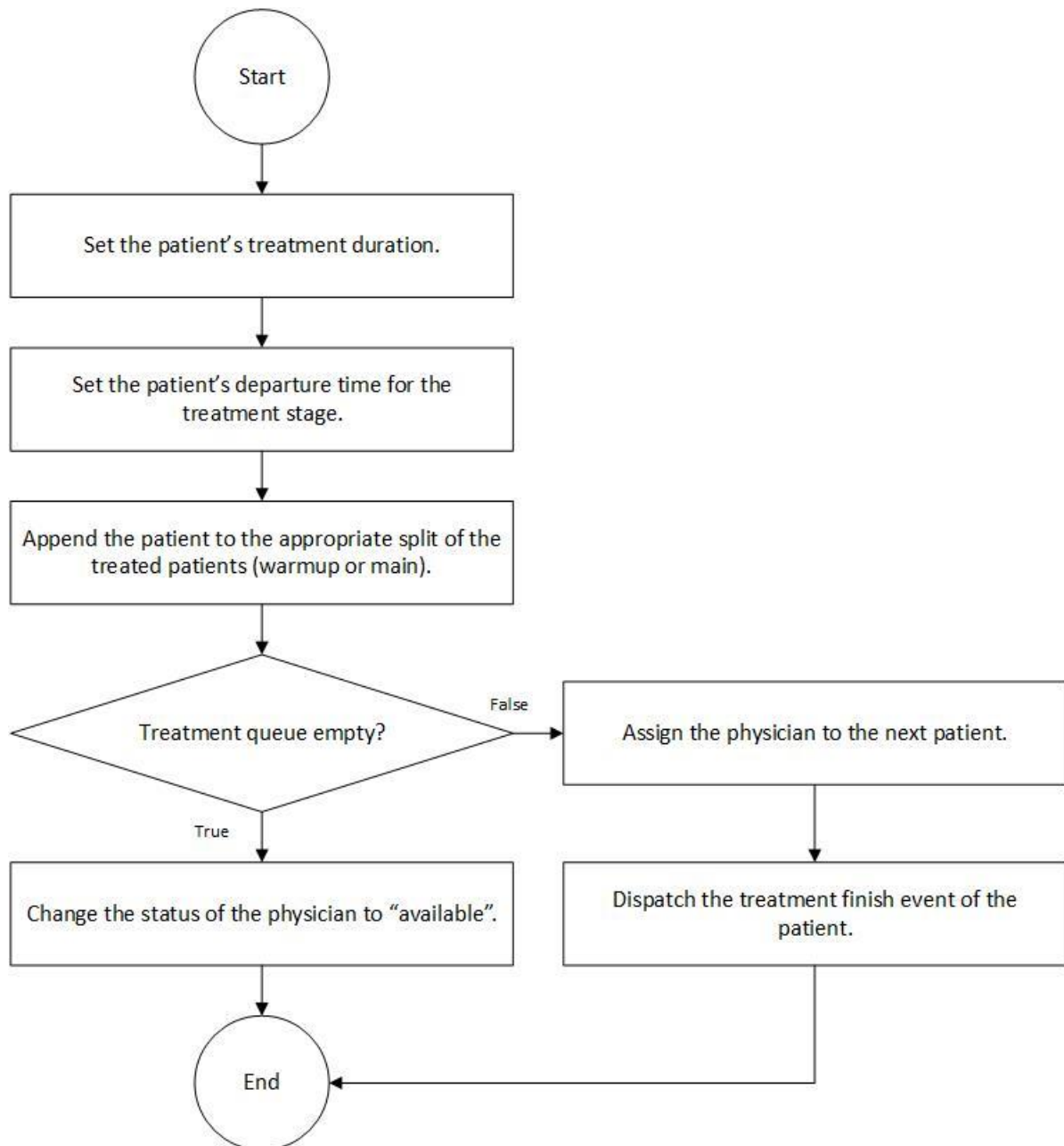
فرایند با تنظیم مدت زمان معاینه بیمار آغاز می‌شود. سپس زمان خروج بیمار از مرحله معاینه تنظیم می‌شود و بعد از آن زمان ورود بیمار به مرحله درمان تعیین می‌شود. در مرحله بعد، یک پزشک موجود به صورت تصادفی پیدا می‌شود. اگر صف درمان خالی نباشد یا پزشک موجود نباشد، بیمار به صف درمان اضافه می‌شود. در غیر این صورت، پزشک انتخاب شده به بیمار اختصاص داده می‌شود و وضعیت پزشک به "مشغول" تغییر می‌کند. سپس رویداد پایان درمان بیمار اعزام می‌شود.

بعد از این مرحله، اگر صف معاینه با اولویت بالا خالی باشد، سیستم بررسی می‌کند که آیا صف معاینه با اولویت پایین نیز خالی است یا خیر. اگر هر دو صف خالی باشند، وضعیت پزشک به "آماده" تغییر می‌کند و فرایند به پایان می‌رسد. اما اگر هر یک از صف‌ها خالی نباشند، پزشک به بیمار بعدی اختصاص داده شده و رویداد پایان درمان بیمار اعزام می‌شود.



اتمام درمان

فرایند با تنظیم مدت زمان درمان بیمار آغاز می‌شود و سپس زمان خروج بیمار از مرحله درمان تنظیم می‌گردد. بیمار به قسمت مناسب از بیماران درمان‌شده (گرم شدن یا اصلی) اضافه می‌شود. در ادامه، سیستم بررسی می‌کند که آیا صف درمان خالی است یا خیر. اگر صف درمان خالی باشد، وضعیت پزشک به "آماده" تغییر می‌کند و فرایند به پایان می‌رسد. اما اگر صف درمان خالی نباشد، پزشک به بیمار بعدی اختصاص داده شده و رویداد پایان درمان بیمار اعزام می‌شود.



نتایج

شبیه سازی برای ۳۰ روز ۲۴ ساعته به همراه یک دوره راه اندازی ۲ روزه است. برای افزایش دقت عددی نتایج، شبیه سازی را ۵۰ بار اجرا می کنیم. اعداد گزارش شده در ادامه، میانگین مقادیر به دست آمده در ۵۰ بار شبیه سازی است.

راه اندازی

نتایج به دست آمده در دوره راه اندازی به صورت زیر است:

مقدار	آماره
0.816000	میانگین درصد اشتغال پزشکان معاینه گر
0.548436	میانگین درصد اشتغال پزشکان درمان گر
142.900000	تعداد کل بیماران
0.273416	میانگین درصد انتظار
1.980000	بیشترین طول صف بیماران اورژانسی مرحله معاینه
0.006251	میانگین طول صف بیماران اورژانسی مرحله معاینه
0.401133	درصد بیماران اورژانسی بدون معطلی مرحله معاینه
0.185962	درصد بیماران اورژانسی کم معطلی (زیر ۴ دقیقه) مرحله معاینه
4.860000	بیشترین طول صف بیماران غیراورژانسی مرحله معاینه
0.317147	میانگین طول صف بیماران غیراورژانسی مرحله معاینه
0.423251	درصد بیماران غیراورژانسی بدون معطلی مرحله معاینه
0.124247	درصد بیماران غیراورژانسی کم معطلی (زیر ۴ دقیقه) مرحله معاینه
3.220000	بیشترین طول صف مرحله درمان
0.051021	میانگین طول صف مرحله درمان
0.699558	درصد بیماران بدون معطلی مرحله درمان
0.115073	درصد بیماران کم معطلی (زیر ۴ دقیقه) مرحله درمان

اجرای اصلی

نتایج به دست آمده در شبیه سازی اصلی به صورت زیر است:

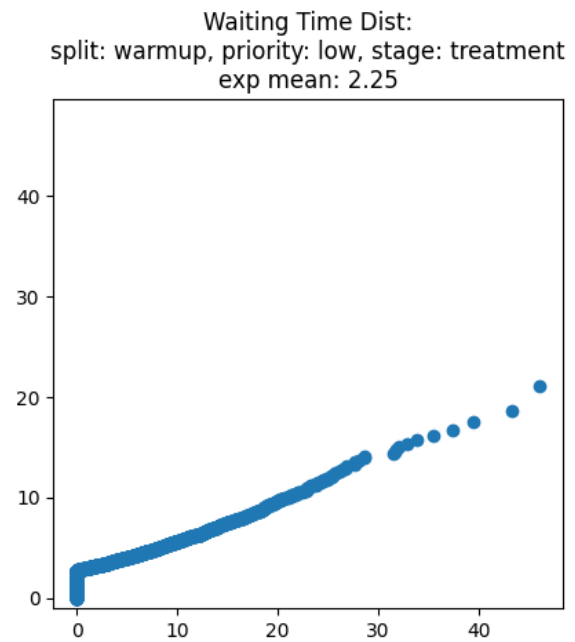
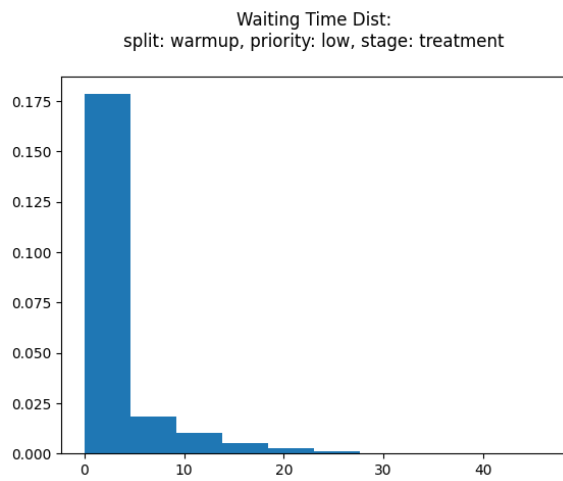
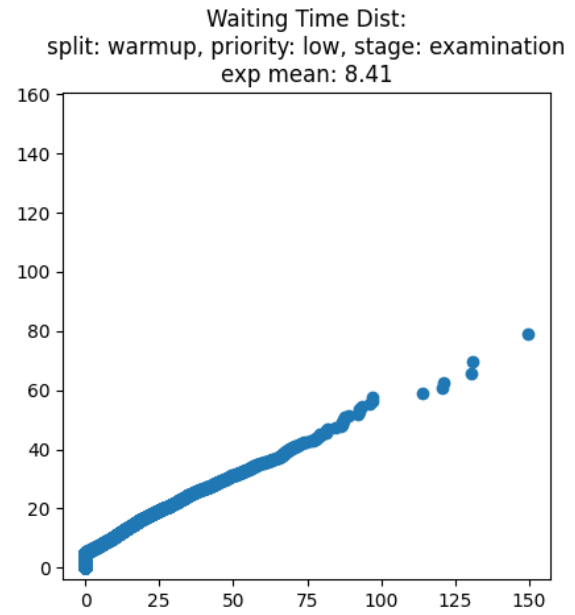
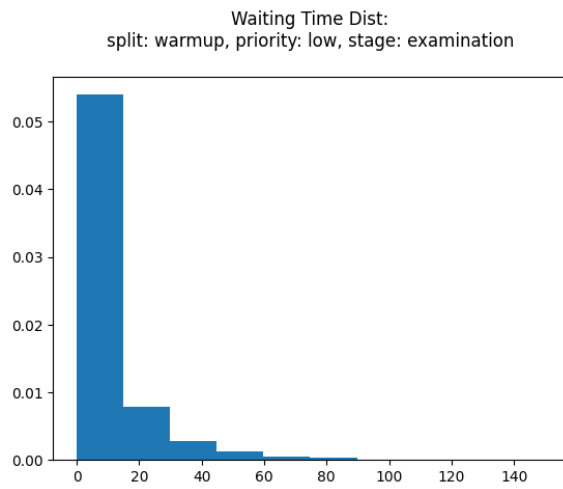
مقدار	آماره
0.819477	میانگین درصد اشتغال پزشکان معاینه گر
0.554320	میانگین درصد اشتغال پزشکان درمان گر
2160.060000	تعداد کل بیماران
0.287117	میانگین درصد انتظار
1.980000	بیشترین طول صف بیماران اورژانسی مرحله معاینه
0.006251	میانگین طول صف بیماران اورژانسی مرحله معاینه
0.430730	درصد بیماران اورژانسی بدون معطلی مرحله معاینه
0.165075	درصد بیماران اورژانسی کم معطلی (زیر ۴ دقیقه) مرحله معاینه
4.860000	بیشترین طول صف بیماران غیراورژانسی مرحله معاینه
0.317147	میانگین طول صف بیماران غیراورژانسی مرحله معاینه
0.428526	درصد بیماران غیراورژانسی بدون معطلی مرحله معاینه
0.112624	درصد بیماران غیراورژانسی کم معطلی (زیر ۴ دقیقه) مرحله معاینه
3.220000	بیشترین طول صف مرحله درمان
0.051021	میانگین طول صف مرحله درمان
0.696804	درصد بیماران بدون معطلی مرحله درمان
0.105609	درصد بیماران کم معطلی (زیر ۴ دقیقه) مرحله درمان

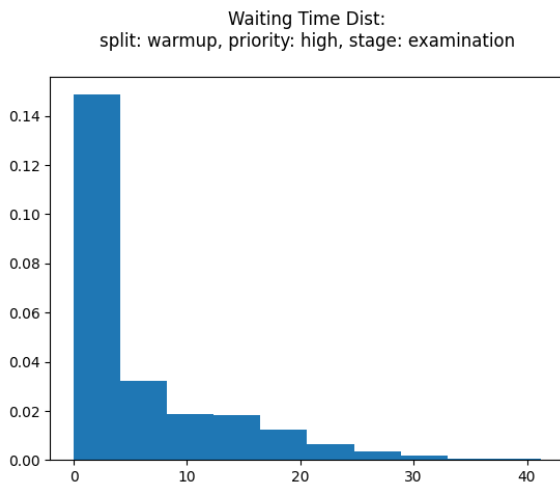
توزیع‌ها

برای هر فاز شبیه‌سازی (گرم‌کردن، اجرای اصلی)، و به تفکیک مرحله (معاینه/درمان) و نوع بیمار (اورژانسی/غیراورژانسی)، نمودارهای مدت زمان انتظار زیر به دست می‌آید. نمودارهای سمت چپ جدول، نمودار اصلی، و نمودارهای سمت راست، نمودارهای QQ Plot هستند. توزیع انتخابی، توزیع نمایی است که با محاسبه چنک‌ها و شکل نمودارهای سمت راست، می‌توان نتیجه‌گیری کرد که انتخاب درستی بوده چرا که خط به دست آمده نزدیک به نیمساز ناحیه اول است.

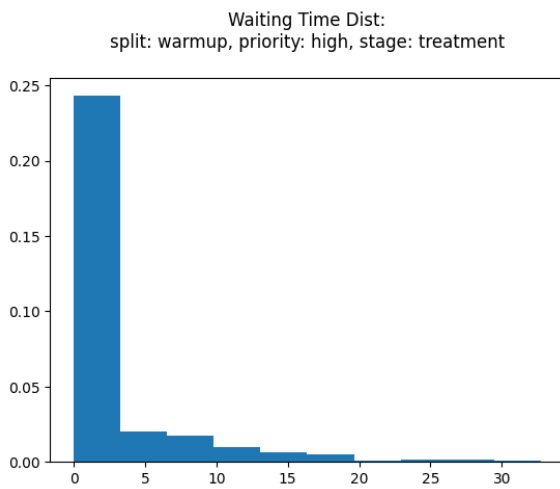
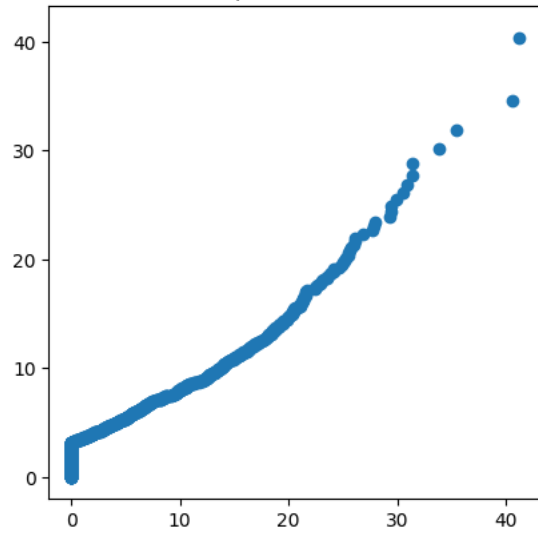
هر چنک از رابطه‌ی زیر به دست می‌آید:

$$q_i = -mean \times e^{1 - \frac{i+0.5}{n}}$$

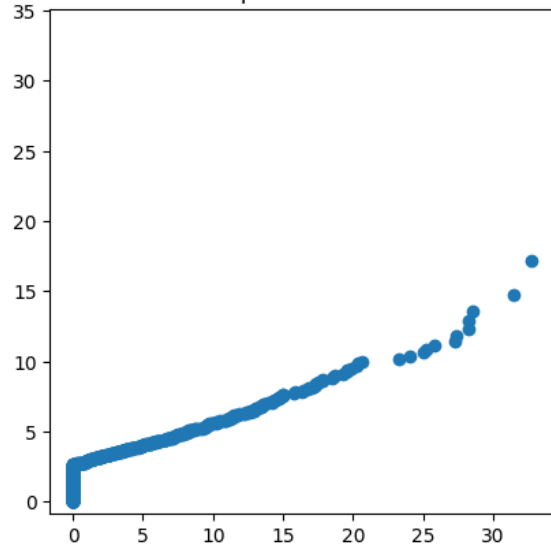


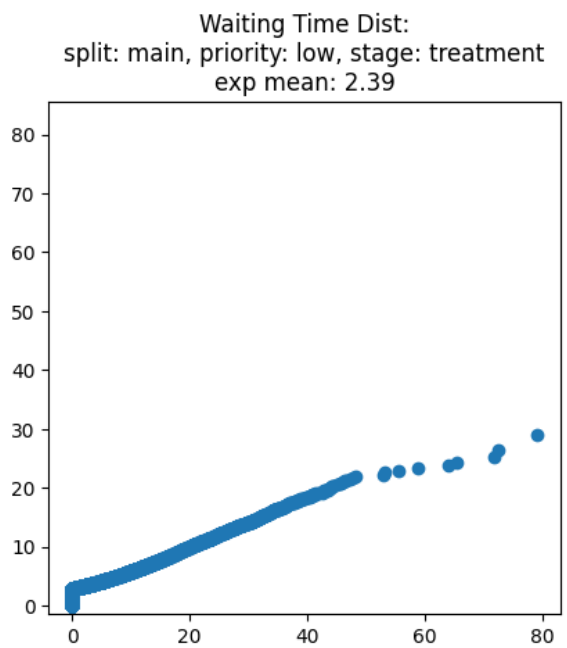
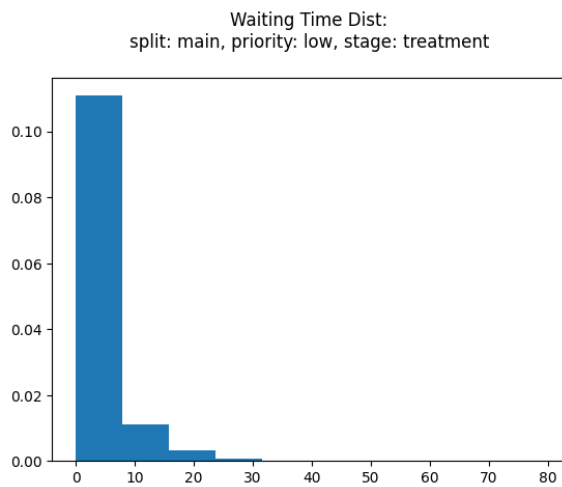
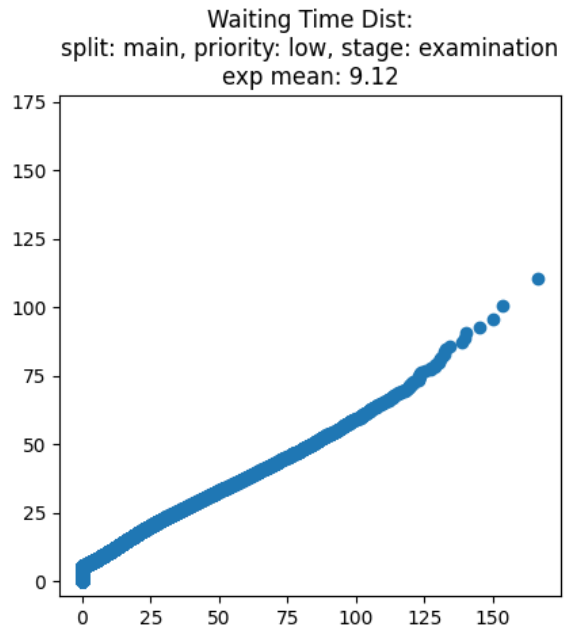
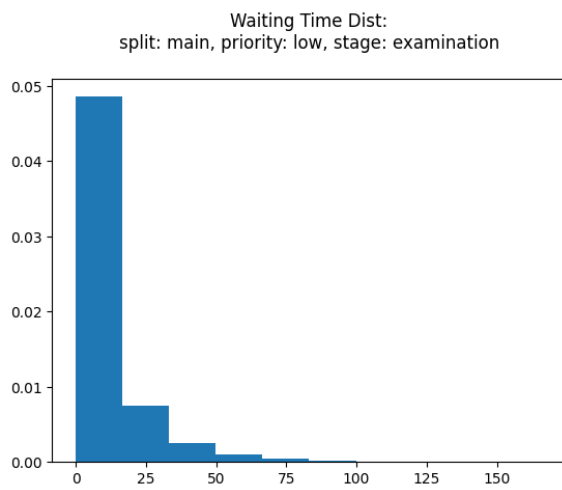


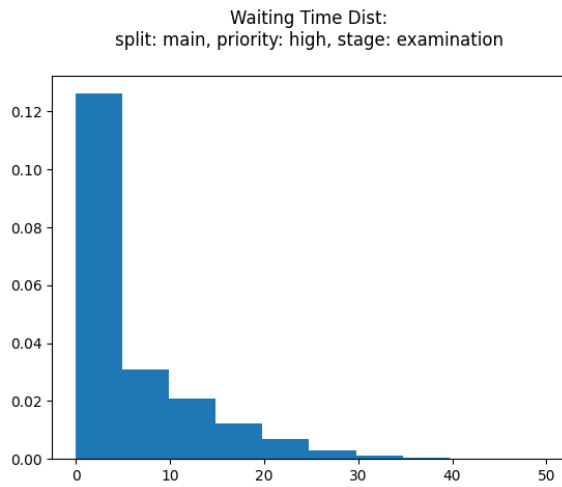
Waiting Time Dist:
split: warmup, priority: high, stage: examination
exp mean: 5.25



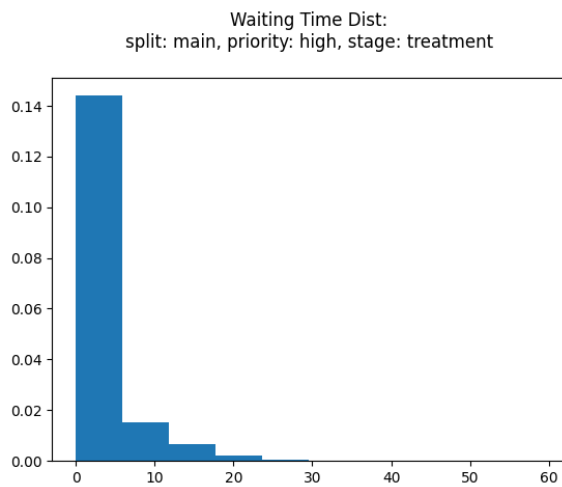
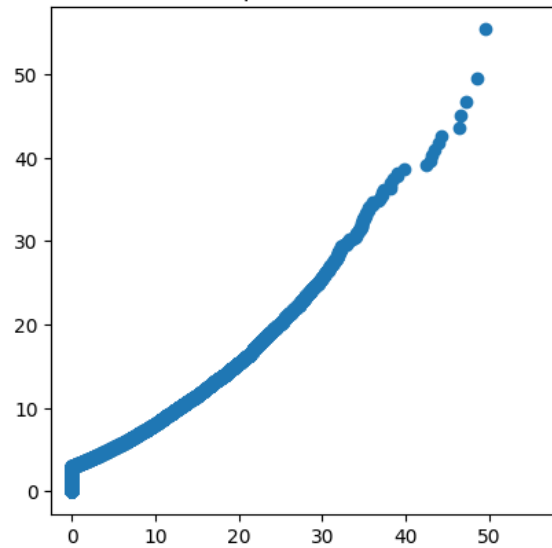
Waiting Time Dist:
split: warmup, priority: high, stage: treatment
exp mean: 2.24



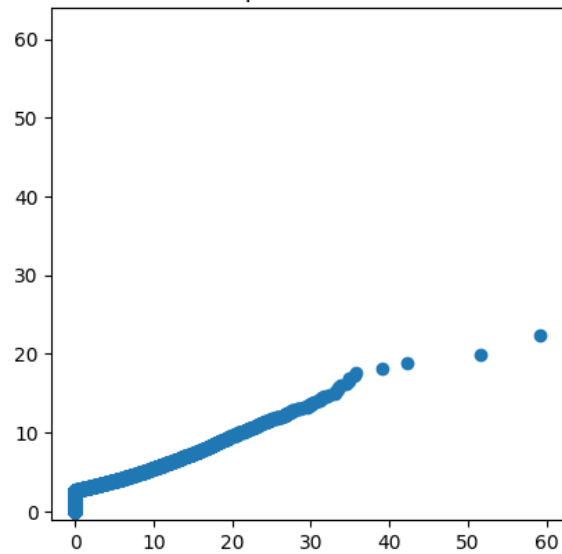




Waiting Time Dist:
split: main, priority: high, stage: examination
exp mean: 5.33



Waiting Time Dist:
split: main, priority: high, stage: treatment
exp mean: 2.15



تعداد بهینه پزشکان

برای به دست آوردن تعداد بهینه‌ی پزشکان، یک معیار امتیاز با رابطه‌ی زیر تعریف می‌کنیم:

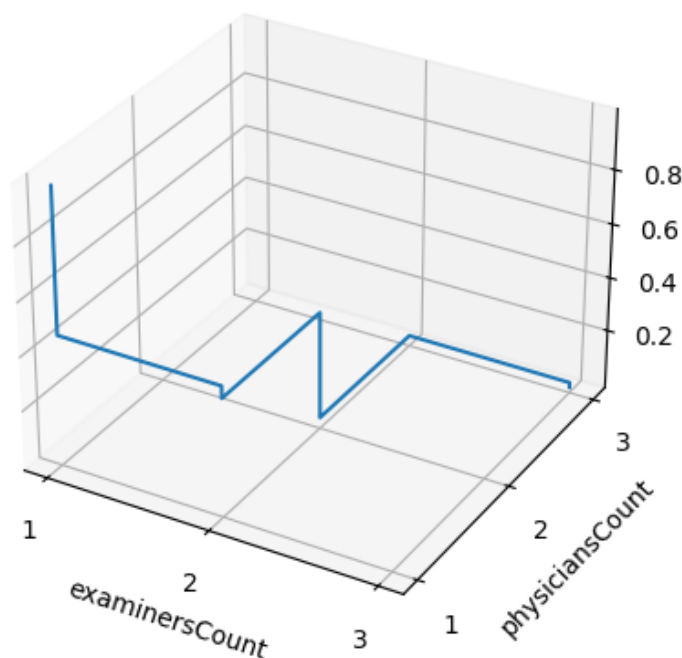
$$\text{score} = \alpha(\text{average utilization percentage}) + (1 - \alpha)(1 - \text{average waiting percentage})$$

شبیه‌سازی را به تعداد ۵۰ بار برای هر زوج (تعداد پزشک درمانگر، تعداد پزشک معاینه‌گر) اجرا کرده، و درصد‌های خدمت‌دهی و انتظار را به دست می‌آوریم. تعداد هر نوع پزشک بین ۱ تا ۳ متغیر است و در نتیجه ۹ ترکیب مختلف داریم و امتیاز هر حالت را به ازای آلفاهای متفاوت محاسبه می‌کنیم:

تعداد پزشکان معاینه‌گر	تعداد پزشکان درمانگر	میانگین درصد اشتغال	میانگین درصد انتظار	امتیاز (آلفا=۰/۲)	امتیاز (آلفا=۰/۴)	امتیاز (آلفا=۰/۶)	امتیاز (آلفا=۰/۸)
۱	۱	0.686930	0.283851	0.710305	0.704461	0.698618	0.692774
	۲	0.4528201	0.234258	0.74234	0.642725	0.581217	0.519709
	۳	0.342691	0.237104	0.678855	0.594814	0.510773	0.426732
۲	۱	0.458467	0.084746	0.823897	0.732539	0.641182	0.549824
	۲	0.343806	0.002386	0.866852	0.736091	0.605329	0.474567
	۳	0.274620	0.000915	0.854192	0.709299	0.564406	0.419513
۳	۱	0.343409	0.083822	0.801624	0.687070	0.572517	0.457963
	۲	0.274319	0.001573	0.853606	0.708784	0.563962	0.419141
	۳	0.228843	0.000003	0.845766	0.691535	0.537304	0.383074

حال، اگر امتیاز را با مقادیر مختلف آلفا (با گام‌هایی به اندازه 0.001) محاسبه کنیم، خواهیم داشت:

Optimal Number of Doctors With a Given Alpha



مقادیر بیشتر آلفا، به بیشینه‌سازی درصد به‌کارگیری، و مقادیر کمتر آن به کمینه‌سازی زمان انتظار اولویت می‌دهند. با توجه به مقدار آلفای انتخابی، می‌توان از روی نمودار تعداد بهینه پزشکان را پیدا کرد. به عنوان مثال اگر آلفا را برابر ۰/۵ در نظر بگیریم، تعداد ۲ پزشک درمانگر و ۲ پزشک معاینه‌گر بهینه خواهد بود.

نتیجه‌گیری

از نتایج به دست آمده در ۵۰ بار اجرای شبیه‌سازی، می‌توان به ماکسیمم طول صف و میانگین طول صف اشاره کرد. هر دو عدد مقدار کمی دارند که نشان دهنده این است که صف‌های طولانی پدید نمی‌آید. همچنین درصد بیماران بدون معطلی که مستقیماً خدمت درمانی ارائه کردند عدد بالایی را نشان می‌دهد که بیانگر عملکرد بسیار خوب سیستم است. علاوه بر این، با مقایسه‌ی زمان‌های انتظار و طول صف بیماران اورژانسی و بیماران غیراورژانسی، مشاهده می‌شود که سیستم در اولویت دادن به بیماران اورژانسی درست عمل کرده است. در کل، می‌توان نتیجه گرفت که این سیستم قابل اطمینان است.