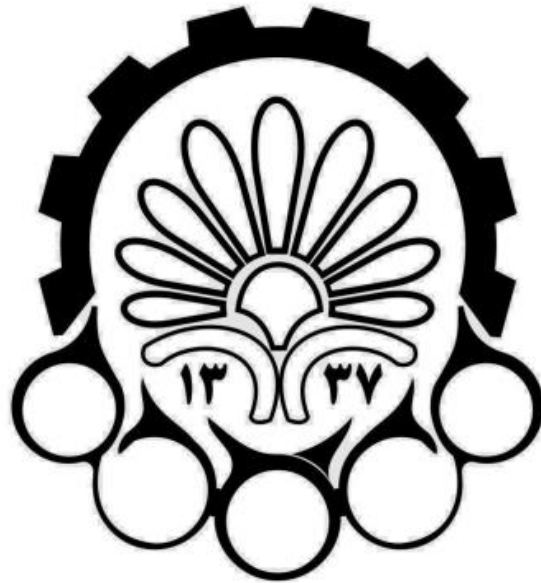


به نام خدا



اصول شبیه سازی

گزارش از روند حل پروژه‌ی پایانی

استاد درس

جناب آقای دکتر احمدی

تهیه کنندگان

فاطمه نیک خواه - ۸۸۲۸۰۹۶

علی لشگری - ۸۷۲۵۷۱۷

محمدرضا تأثیری - ۸۸۱۲۰۷۶

*[This page intentionally left blank]*

# github

SOCIAL CODING



Fork Source Code on Github

<https://github.com/taesiri/Simulation>

We do love Open Source, Full Source code and project documentation is available on [github.com/taesiri/Simulation](https://github.com/taesiri/Simulation).

## فهرست مطالب

۵.....	اطلاعات فنی
۶.....	متن مسئله
۸.....	فرض های ساده ساز
۱۰.....	نکاتی در مورد ساخت اعداد تصادفی
۱۲.....	نمودارهای جریان
۱۷.....	نکاتی در مورد نرم افزار
۱۸.....	شبیه سازی خودکار
۲۲.....	شبیه سازی ۳ بعدی
۲۳.....	عناصر شبیه سازی
۲۹.....	ابزار مطالعه آماری
۳۰.....	نمودارهای جریان مربوط به شبیه سازی
۳۱.....	Class Diagrams

### HELIX 3D TOOLKIT

Helix 3D Toolkit is a collection of custom controls and helper classes for WPF.

License: The MIT License (MIT)

Project page:

<http://helixToolkit.codeplex.com>

Source:

<https://hg.codeplex.com/helixtoolkit>

Build requirements

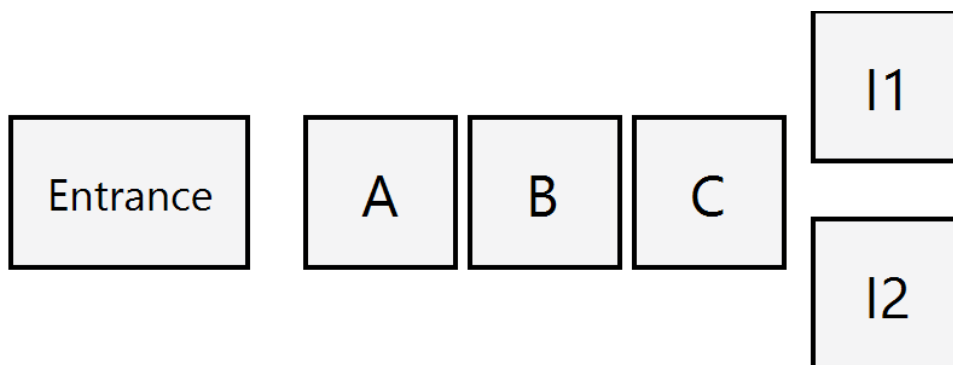
- Microsoft .NET 4.5 Client profile
- Visual Studio 2012

همانند پروژه های قبلی، از بستر نرم افزاری *NET*.  
*Framework 4* برای شبیه سازی استفاده شده است.  
زبان اصلی استفاده شده *C#* است ولی در پشت صحنه  
عملیاتی از زبان برنامه نویسی *R* نیز برای مطالعات آماری  
استفاده شده است.

در این پروژه از کتابخانه ی *Helix 3D Toolkit* برای  
شبیه سازی سه بعدی استفاده کرده ایم.

## متن مسئله

سیستمی شامل یک ایستگاه ورودی، سه ایستگاه کاری A, B, C و یک ایستگاه بازرسی I می باشد، که به صورت سری پشت سر هم قرار گرفته اند؛ علاوه بر این ایستگاه ها، سیستم دارای رباتی است که عملیات انتقال قطعات را انجام می دهد؛ در شکل زیر مکان و چگونگی استقرار ایستگاه ها و ربات نشان داده شده است.



ابتدا مواد اولیه بر اساس یک توزیع  $\exp(12)$  (نمایی با میانگین ۱۲ دقیقه) وارد سیستم می شوند. در این لحظه دو حالت ممکن است رخ دهد. اگر ربات بی کار باشد و ایستگاه A خالی باشد، ربات قطعه را برداشته و بر اساس یک توزیع  $\exp(0.7)$  در ماشین A بارگذاری می نماید و پس از آن ربات آزاد می گردد. در صورت مشغول بودن ربات یا اشغال بودن ایستگاه A، قطعه ورودی، وارد یک صف انتظار FIFO در ایستگاه ورودی می شود؛ پس از اتمام بارگذاری هر یک از قطعات در ایستگاه A، این ایستگاه با یک توزیع  $\exp(1.1)$  شروع به خدمت دهی به قطعه می نماید.

با اتمام کار قطعه در ایستگاه A و در صورت خالی بودن ایستگاه بعد، ربات در صورت آزاد بودن ابتدا عمل بار برداری از ایستگاه A را با توزیع  $\exp(0.7)$  انجام داده و سپس قطعه را در

ایستگاه B با توزیع  $\exp(0.7)$  بارگذاری می نماید. در این لحظه ایستگاه B کار بر روی قطعه را با توزیع  $\exp(1.3)$  آغاز می کند.

پس از اتمام کار قطعه در ایستگاه B و در صورت خالی بودن ایستگاه بعد، ربات در صورت آزاد بودن عملیات بار برداری از ایستگاه B را با توزیع  $\exp(0.7)$  انجام داده و قطعه را در ایستگاه C با توزیع  $\exp(0.7)$  بارگذاری می نماید. در این لحظه ایستگاه C کار بر روی قطعه را با توزیع  $\exp(1.2)$  آغاز می کند و پس از انجام این عملیات، قطعه آماده می شود.

قطعه پردازش شده در ایستگاه C، پس از باربرداری توسط ربات با توزیع  $\exp(0.7)$  از این ایستگاه در ایستگاه بازرسی (I) با توزیع  $\exp(0.7)$  بارگذاری می شود؛ در این ایستگاه دو خدمت دهنده وجود دارند که با توزیع  $\text{Normal}(2,1)$  به خدمت‌دهی می پردازند، در صورت مشغول بودن هر دو خدمت دهنده؛ قطعه وارد شده به این ایستگاه، وارد یک صف انتظار FIFO می شود.

پس از مرحله بازرسی، قطعه از سیستم خارج می شود.

## فرض های ساده ساز

برای حل بهتر این مسئله، گروه فرض های ساده سازی برای این شبیه سازی در نظر گرفته است.

این فرض ها عبارت اند از:

### ۱- عملیات بارگذاری و باربرداری

- عملیات بارگذاری و باربرداری را تبدیل به عملیات انتقال و آماده سازی کرده ایم، به این نحو که ربات ابتدا جعبه را بین دو ایستگاه جابه جا می کند، بعد از انتقال جعبه، مدتی را صرف آماده سازی جعبه در آن ایستگاه می کند.

- بین ایستگاه ورودی و ایستگاه A هر دو نوع عملیات را لحاظ کرده ایم.  
- وقتی کار قطعه در ایستگاه C تمام می شود، فقط عملیات انتقال جعبه به ایستگاه بازرسی (I) را داریم و بارگذاری در این ایستگاه وجود ندارد.

بدیهی است که این فرض ها فقط جهت زیبایی شبیه سازی سه بعدی انجام شده است.

### ۲- اولیت بندی انتقال

مشاهده نشان می دهد که امکان دارد رویدادهای اتمام سرویس دهی، ممکن است برای چند ایستگاه همزمان رخ دهد؛ یعنی در چند ایستگاه به طور همزمان نیاز به ربات برای انتقال قطعه داشته باشیم. ابتدا به بیان تمامی احتمالاتی که قرار است اتفاق بی افتد می پردازیم.

- کار بر روی قطعه تمام شده است؛ ولی ربات مشغول است



- کار بر روی قطعه تمام شده است؛ ربات آزاد ولی ایستگاه بعدی مشغول عملیات است.

- کار بر روی قطعه تمام شده است؛ ربات مشغول، و همچنین ایستگاه بعدی مشغول عملیات است.

همیشه ابتدا از چپ به راست عملیات نیاز به ربات را بررسی می کنیم! یعنی پس از این که ربات آزاد شد، اول به ایستگاه C رسیدگی می شود که آیا نیاز به ربات دارد یا نه؛ پس از آن ایستگاه B و بعد از آن A .

۳- انتقال از ایستگاه C به ایستگاه بازرسی

در ایستگاه C دو کارگر آماده خدمت دهی هستند. اولیت بندی بین این دو خدمت دهنده صورت نگرفته است. یعنی هر بار به صورت تصادفی یک عدد بین ۱ تا ۱۰ تولید شده، اگر عدد از ۵ کمتر بود، جعبه به کارگر اول انتقال می یابد، در غیر این صورت جعبه به کارگر دوم انتقال داده می شود.

۴- زمان "Warm up time" برابر یک ساعت در نظر گرفته شده است. درحالتی که جعبه قبل از ساعت ۹:۰۰ صبح وارد شده است، ولی زمان خروج آن پس از ساعت ۹:۰۰ صبح هست، جعبه را در محاسبات شبیه سازی به حساب می آوریم.

## نکاتی در مورد ساخت اعداد تصادفی

برای تولید اعداد تصادفی، در این شبیه سازی از روش های تبدیل مستقیم و تبدیل معکوس استفاده شده است. ابتدا از روش تبدیل مستقیم برای تولید اعداد تصادفی با توزیع نرمال در بازه ی  $[0,1]$  استفاده شده است. سپس اعداد بدست آمده را به روش تبدیل معکوس به اعداد تصادفی با توزیع نمایی تبدیل می کنیم.

به فرمول های زیر توجه فرمایید.

روش تبدیل مستقیم

$$Z_1 = \sqrt{-2\ln(R_1)} * \cos(2\pi R_2)$$

$$Z_2 = \sqrt{-2\ln(R_2)} * \sin(2\pi R_1)$$

روش تبدیل معکوس (نمایی)

$$F(X) = 1 - e^{-\lambda X} = R \Rightarrow$$

$$e^{-\lambda X} = 1 - R \Rightarrow -\lambda X = \ln(1 - R)$$

$$\Rightarrow X = -\frac{1}{\lambda} \ln(1 - R) \sim -\frac{1}{\lambda} \ln(R)$$

توجه داشته باشید که

$$\lambda = \frac{1}{\bar{X}}$$

به طور مثال، اگر بخواهیم اعدادی با میانگین ۱۲ بسازیم؛ از فرمول زیر

$$X = -12\ln(1 - R)$$

استفاده می کنیم.

نکته : اعداد تصادفی اولیه برای روش تبدیل مستقیم را از کتابخانه درونی C# می سازیم.

به صورت زیر:

```
var rand = new Random(DateTime.Now.Millisecond);  
int n = rand.Next(10, 100);
```

## نمودارهای جریان

از آنجا که نمودارهای جریان این پروژه را از طریق خود برنامه نیز قابل مشاهده می باشند و به دلیل تعدد این نمودارها فقط به قراردادن تعدادی از آنها بسنده می کنیم.

### Events

New Box Arrival

Service Started at Station A  
Service Started at Station B  
Service Started at Station C

Service Ended at Station A  
Service Ended at Station B  
Service Ended at Station C

Box Moved to Station A  
Box Moved to Station B  
Box Moved to Station C

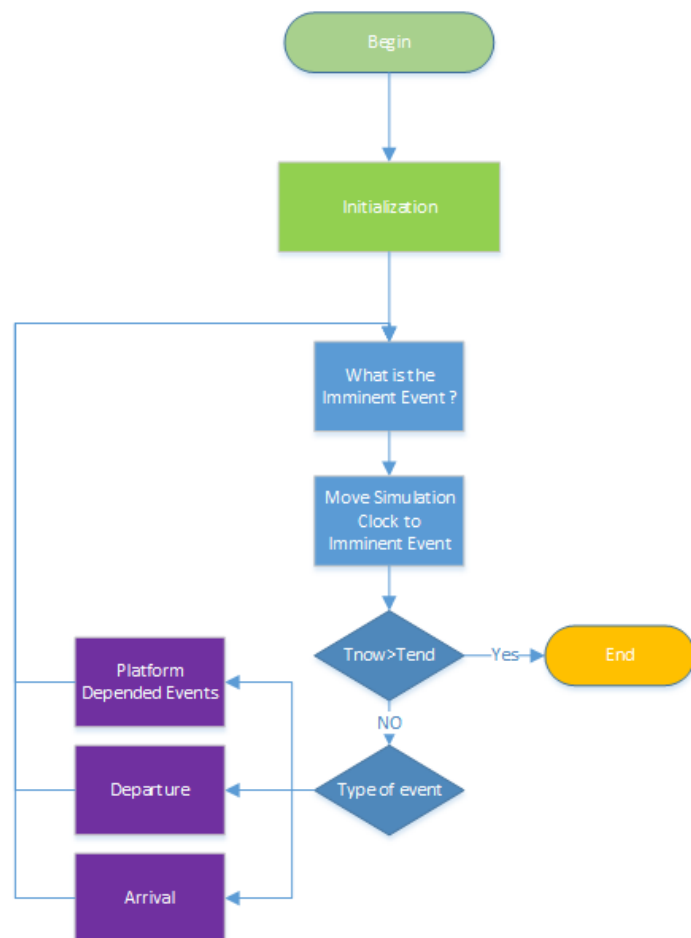
Robot Job Finished

Robot

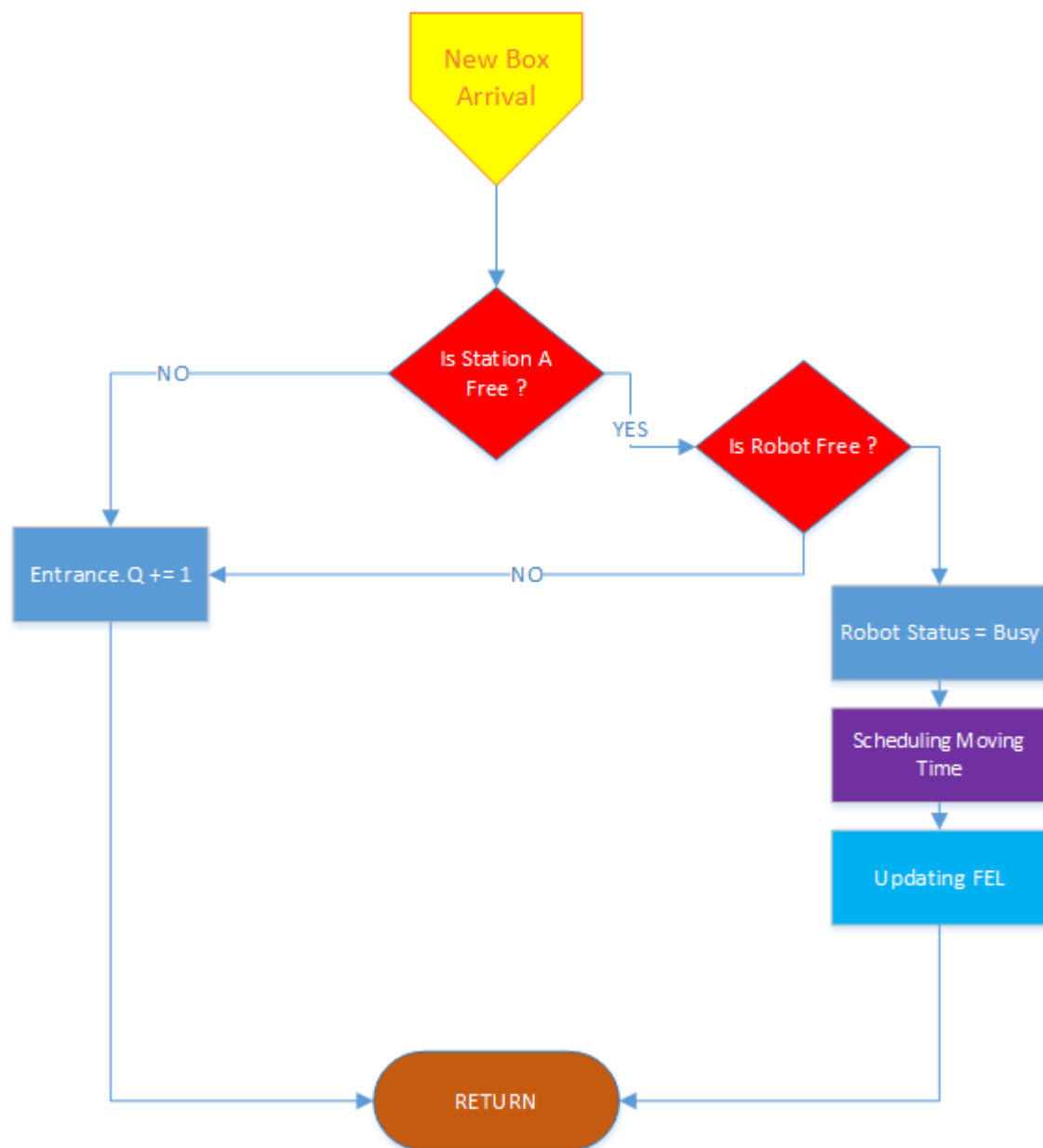
```
{
  Box Loaded On Station A
  Box Loaded On Station B
  Box Loaded On Station C
}
```

Box Moved to Inspector  
Inspector 1 Job Done  
Inspector 2 Job Done

Box left the System

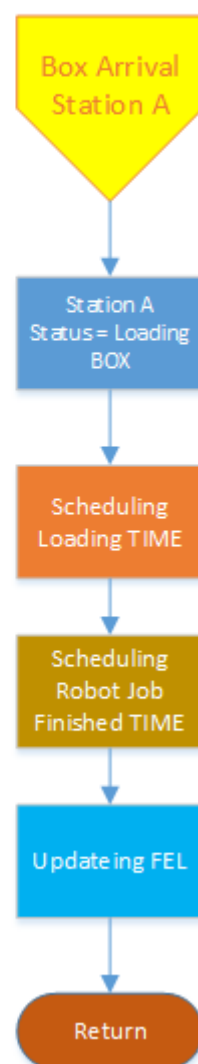


کنترلر شبیه سازی



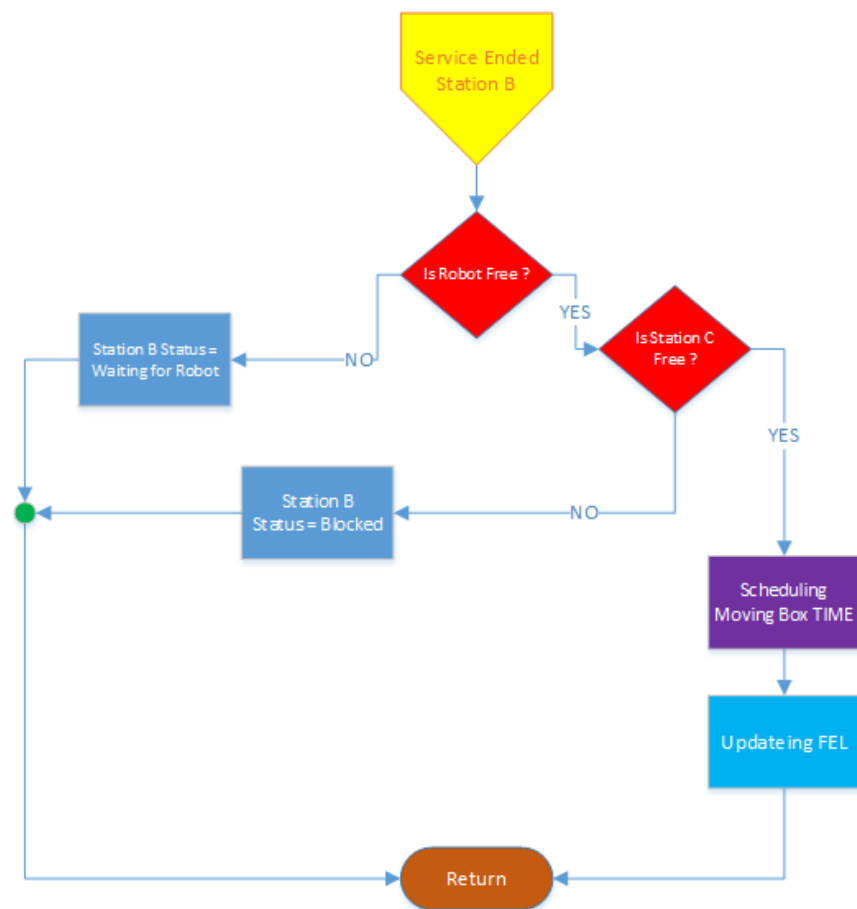
رویداد ورود جعبه‌ی جدید

BOX Arrived to Station A



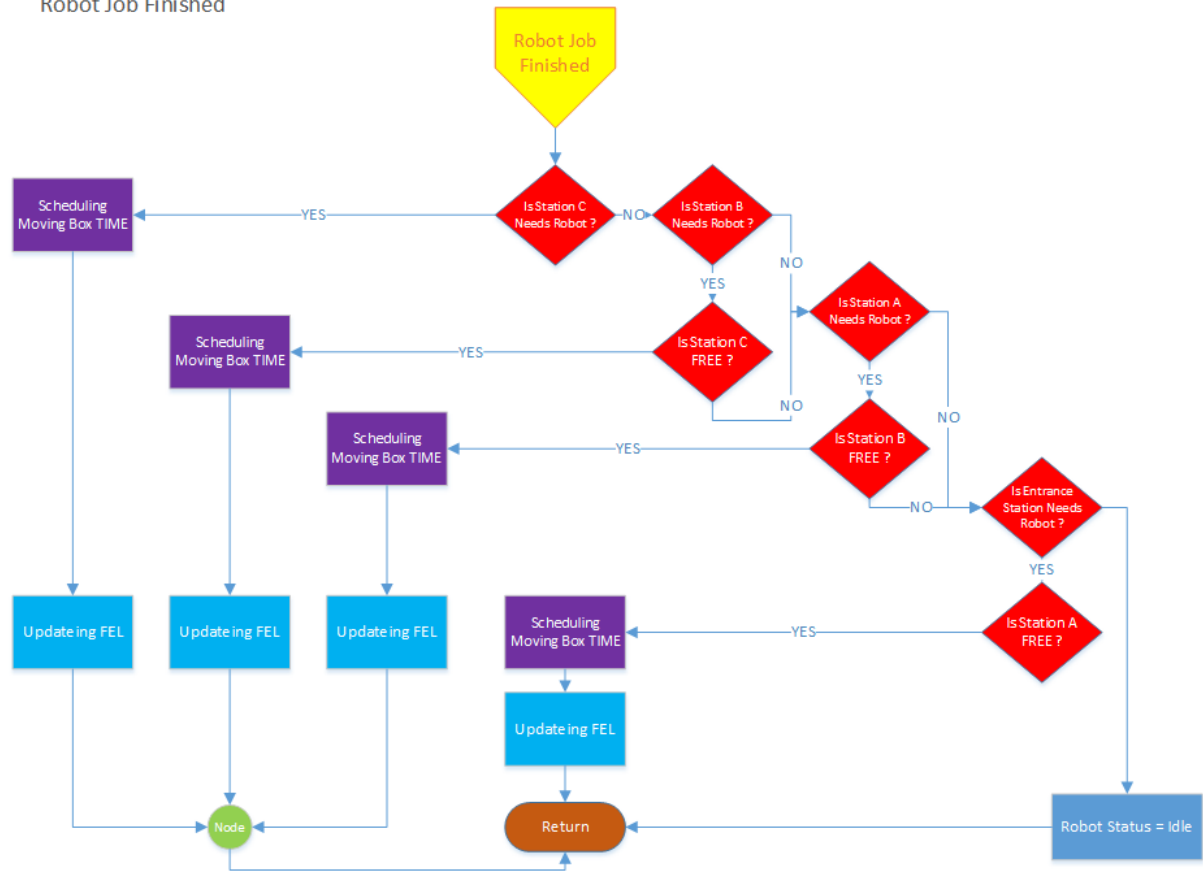
ورود جعبه به ایستگاه A

## Service Finished at Station B



اتمام سرویس در ایستگاه B

Robot Job Finished



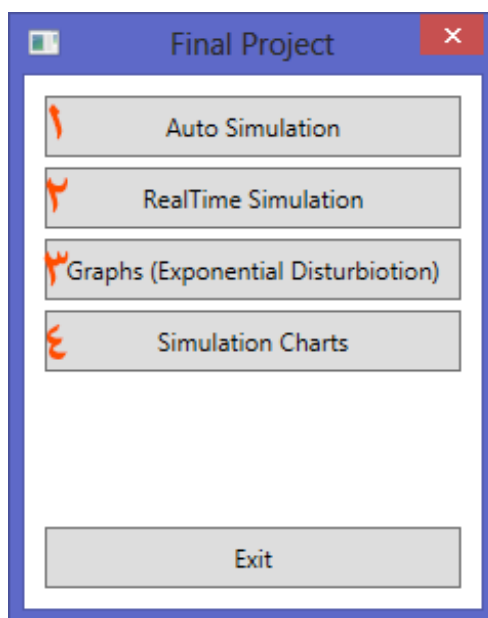
رویداد اتمام کار ربات



## نکاتی در مورد نرم افزار

به دلیل این که نحوه پیاده سازی پروژه‌ی نهایی، کاملاً متفاوت با پروژه‌های قبلی است، گزارش آن نیز متفاوت است. در این قسمت به آموزش نرم افزار می‌پردازیم.

در اولین اجرای برنامه صفحه‌ایی مانند زیر خواهید دید.



- ۱- شبیه سازی خودکار برای ۱۰ بار اجرای سیستم ( گزارش کار)
- ۲- شبیه سازی سه بعدی بلادرنگ (Real-time 3D Simulation)
- ۳- ابزار مطالعه آماری برای توزیع نمایی با میانگین دلخواه (گراف)
- ۴- نمودارهای جریان مربوط به شبیه سازی

## شبیه سازی خودکار

در این حالت، برنامه به صورت خودکار، ۱۰ بار شبیه سازی را اجرا کرده و نتایج آن ها را گزارش می دهد.

این گزارش شامل میانگین زمان خدمت دهی، میانگین اشتغال کلیه ایستگاه ها است، زمان شروع و اتمام شبیه سازی، نمودار های مربوط به شبیه سازی و سایر موارد است.

نمونه خروجی برنامه

### Generated Report

#### Final Project - Box Factory

Warm up time: 1 hour - Everything counted and calculated after 9:00 AM

#RUN	Start Date	End Date (Last Departure)	Duration	Total Boxes	Average Time (BOX in System)
1	01/18/2013 08:00:00	01/18/2013 18:26:21	10:26:21	20	77 - Minutes
2	01/18/2013 08:00:00	01/18/2013 16:56:10	08:56:10	18	76 - Minutes
3	01/18/2013 08:00:00	01/18/2013 16:07:37	08:07:37	11	37 - Minutes
4	01/18/2013 08:00:00	01/18/2013 15:57:22	07:57:22	14	55 - Minutes
5	01/18/2013 08:00:00	01/18/2013 16:26:09	08:26:09	13	44 - Minutes
6	01/18/2013 08:00:00	01/18/2013 16:48:21	08:48:21	12	41 - Minutes
7	01/18/2013 08:00:00	01/18/2013 16:30:09	08:30:09	9	33 - Minutes
8	01/18/2013 08:00:00	01/18/2013 17:04:27	09:04:27	14	43 - Minutes
9	01/18/2013 08:00:00	01/18/2013 15:13:49	07:13:49	16	49 - Minutes
10	01/18/2013 08:00:00	01/18/2013 16:24:19	08:24:19	13	45 - Minutes
Total	---	---	---	140	50 - Minutes Per Box

نمونه خروجی برنامه - اطلاعات کلی در مورد هر اجرا

#RUN	Station A Total Service Time	Station B Total Service Time	Station C Total Service Time	Inspector Total Service Time	Total Service Time
1	01:10:00	01:43:00	01:18:00	00:43:15	04:54:15
2	00:52:00	01:47:00	01:10:00	00:39:27	04:28:27
3	00:44:00	00:22:00	00:28:00	00:18:26	01:52:26
4	01:03:00	00:38:00	00:45:00	00:27:09	02:53:09
5	00:57:00	00:54:00	01:05:00	00:32:17	03:28:17
6	00:57:00	01:15:00	00:42:00	00:24:15	03:18:15
7	00:32:00	00:29:00	00:34:00	00:15:15	01:50:15
8	00:56:00	00:51:00	01:04:00	00:33:01	03:24:01
9	01:06:00	00:57:00	01:03:00	00:37:38	03:43:38
10	00:39:00	00:56:00	01:06:00	00:28:42	03:09:42
Average	54 - Minutes	59 - Minutes	56 - Minutes	30 - Minutes	03:18:00

### نمونه خروجی برنامه - مدت خدمت دهی در هر اجرا

#### Detailed Information for Each Run

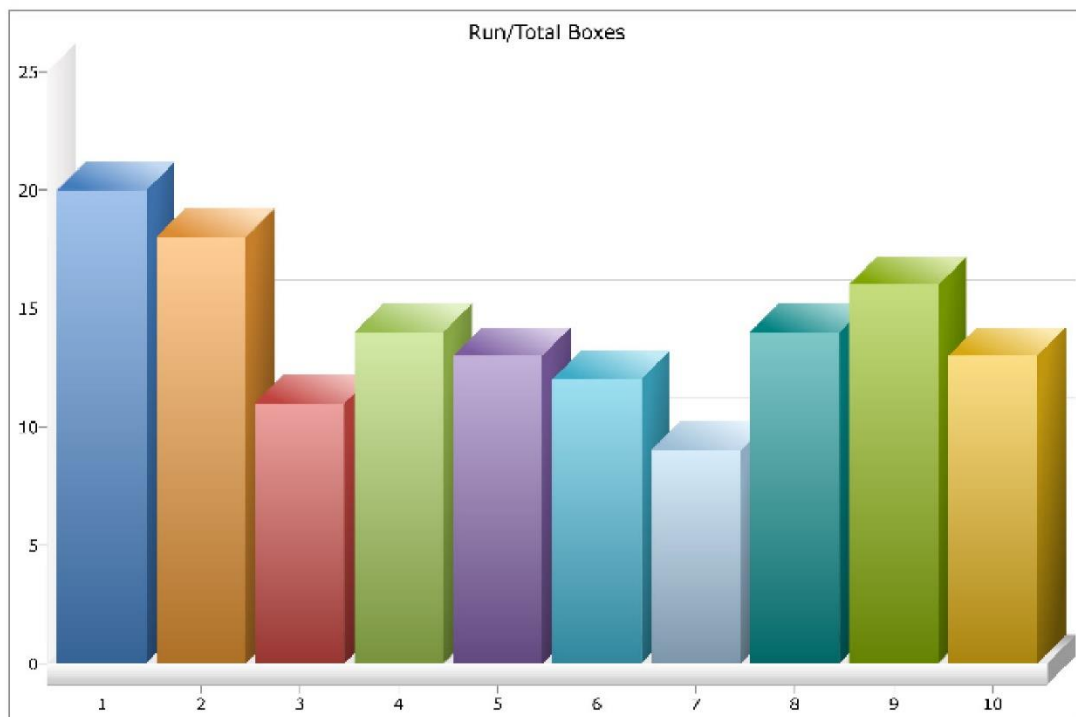
#Run. #Box	Arrival Date	Departure Date	Service Start (A)	Service End (A)	Service Start (B)	Service End (B)	Service Start (C)	Service End (C)	Total Service
1.1	08:31:00	09:16:36	08:40:00	08:48:00	08:58:00	09:07:00	09:09:00	09:10:00	00:22:36
1.2	08:47:00	09:42:53	09:04:00	09:06:00	09:11:00	09:20:00	09:28:00	09:36:00	00:20:53
1.3	08:55:00	09:54:34	09:22:00	09:30:00	09:32:00	09:41:00	09:51:00	09:52:00	00:19:34
1.4	10:18:00	10:56:57	10:24:00	10:32:00	10:38:00	10:47:00	10:50:00	10:51:00	00:19:57
1.5	10:37:00	11:16:23	10:44:00	10:49:00	10:54:00	11:03:00	11:05:00	11:13:00	00:24:23
1.6	12:00:00	12:24:16	12:04:00	12:05:00	12:11:00	12:12:00	12:18:00	12:19:00	00:03:16
1.7	13:23:00	14:01:22	13:25:00	13:33:00	13:39:00	13:44:00	13:51:00	13:54:00	00:19:22
1.8	13:26:00	14:12:24	13:45:00	13:46:00	13:57:00	13:58:00	14:04:00	14:06:00	00:04:24
1.9	13:27:00	14:52:08	14:11:00	14:12:00	14:14:00	14:23:00	14:33:00	14:41:00	00:20:08
1.10	13:30:00	15:01:51	14:20:00	14:21:00	14:39:00	14:40:00	14:52:00	15:00:00	00:10:51
1.11	13:34:00	15:24:34	14:45:00	14:46:00	14:54:00	14:57:00	15:11:00	15:19:00	00:14:34
1.12	13:41:00	15:39:39	15:00:00	15:02:00	15:21:00	15:22:00	15:28:00	15:36:00	00:12:39
1.13	13:50:00	16:06:21	15:30:00	15:31:00	15:37:00	15:46:00	15:50:00	15:51:00	00:12:21
1.14	15:13:00	16:12:00	15:40:00	15:41:00	16:00:00	16:01:00	16:07:00	16:09:00	00:05:00

### نمونه خروجی برنامه - ریز اطلاعات مربوط به هر جعبه در هر بار اجرا

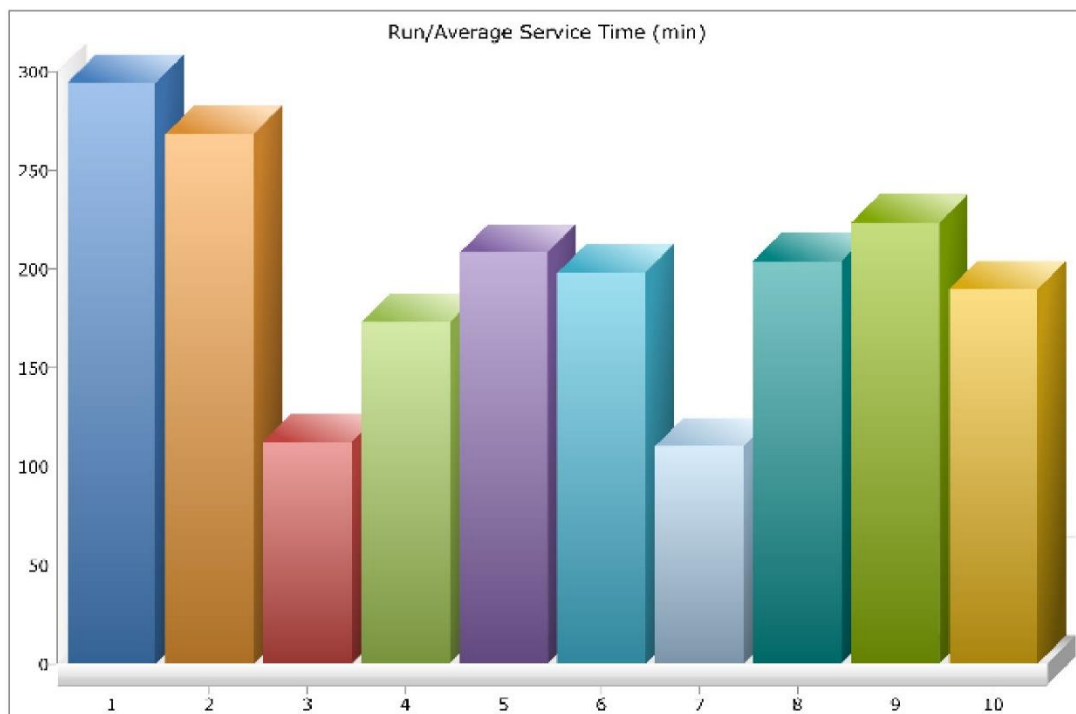
1.15	15:46:00	16:35:57	16:10:00	16:11:00	16:13:00	16:14:00	16:25:00	16:26:00	00:04:57
1.16	15:47:00	16:53:26	16:19:00	16:20:00	16:33:00	16:34:00	16:40:00	16:48:00	00:12:26
1.17	15:48:00	17:27:50	16:50:00	16:51:00	16:58:00	17:02:00	17:14:00	17:18:00	00:11:50
1.18	15:51:00	17:45:26	17:04:00	17:05:00	17:20:00	17:29:00	17:37:00	17:38:00	00:12:26
1.19	15:57:00	17:56:31	17:31:00	17:32:00	17:39:00	17:40:00	17:50:00	17:51:00	00:05:31
1.20	16:00:00	18:26:21	17:53:00	18:01:00	18:07:00	18:16:00	18:22:00	18:23:00	00:20:21
--	--	--	--	--	--	--	--	--	--
	>>		R	U	N		2		
--	--	--	--	--	--	--	--	--	--
2.1	08:09:00	09:15:41	08:44:00	08:45:00	08:47:00	08:56:00	09:06:00	09:08:00	00:14:41
2.2	08:33:00	09:36:08	08:49:00	08:57:00	09:08:00	09:12:00	09:19:00	09:27:00	00:21:08
2.3	08:40:00	09:40:49	09:21:00	09:22:00	09:24:00	09:33:00	09:37:00	09:38:00	00:12:49
2.4	08:43:00	10:06:03	09:34:00	09:42:00	09:44:00	09:53:00	09:56:00	10:04:00	00:26:03
2.5	08:44:00	10:14:33	09:54:00	09:55:00	09:58:00	09:59:00	10:07:00	10:08:00	00:03:33
2.6	08:48:00	10:38:19	10:00:00	10:01:00	10:13:00	10:14:00	10:20:00	10:28:00	00:13:19
2.7	08:49:00	11:02:35	10:26:00	10:27:00	10:30:00	10:39:00	10:47:00	10:55:00	00:20:35
2.8	08:59:00	11:22:21	10:41:00	10:43:00	10:53:00	11:02:00	11:10:00	11:18:00	00:22:21
2.9	09:09:00	11:37:17	10:59:00	11:07:00	11:12:00	11:21:00	11:23:00	11:25:00	00:21:17
2.10	10:32:00	11:54:40	11:14:00	11:15:00	11:30:00	11:32:00	11:45:00	11:47:00	00:07:40
2.11	11:55:00	12:18:16	11:58:00	11:59:00	12:05:00	12:06:00	12:12:00	12:13:00	00:03:16
2.12	13:18:00	13:41:49	13:21:00	13:29:00	13:31:00	13:35:00	13:37:00	13:38:00	00:15:49
2.13	14:41:00	15:11:38	14:51:00	14:53:00	14:56:00	14:57:00	15:07:00	15:08:00	00:04:38

نمونه خروجی برنامه - ریز اطلاعات مربوط به هر جعبه در هر بار اجرا

Simulation Chart



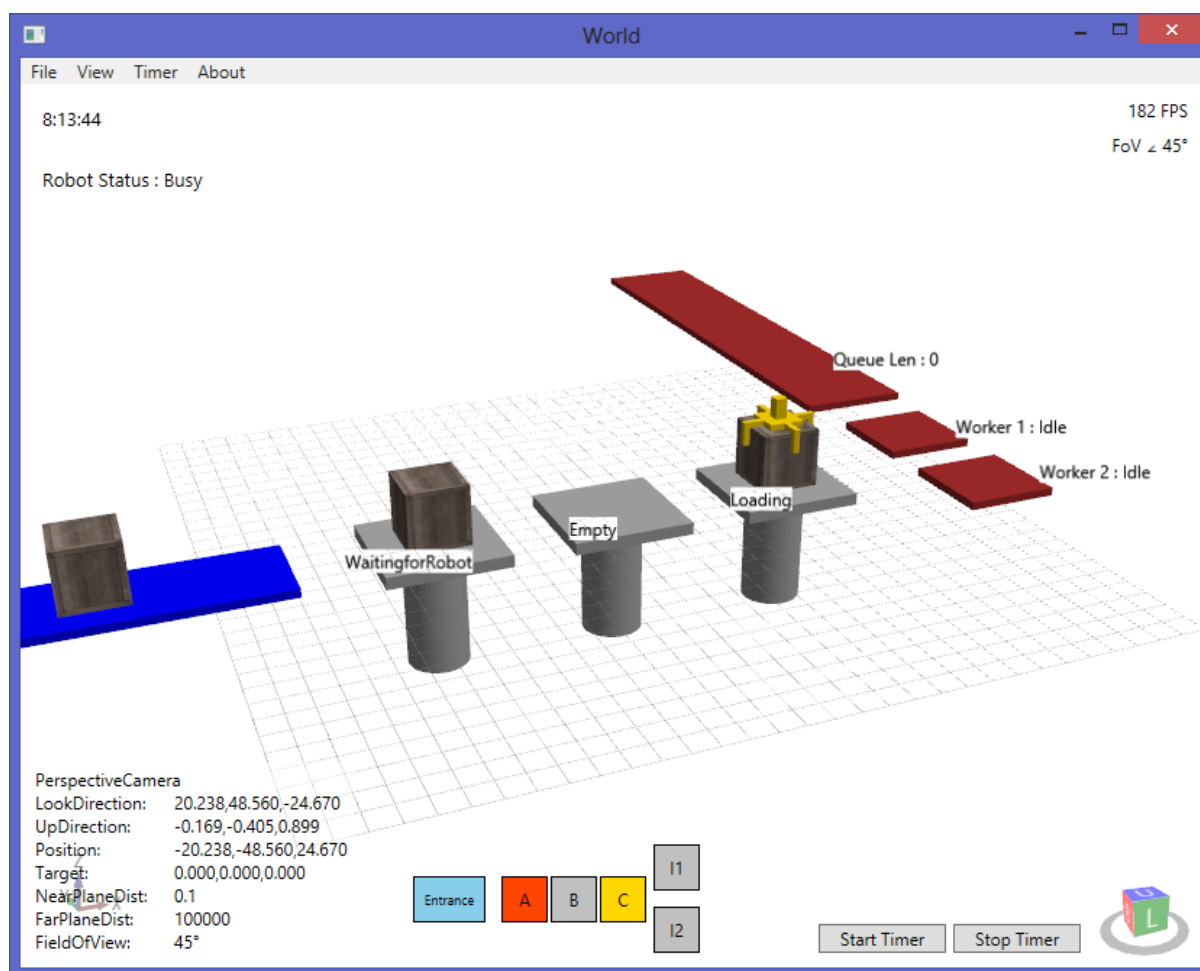
نمونه خروجی برنامه - نمودار مربوط به تعداد جعبه ها در هر اجرا



نمونه خروجی برنامه - نمودار مربوط به متوسط مدت خدمت دهی

## شبیه سازی ۳ بعدی

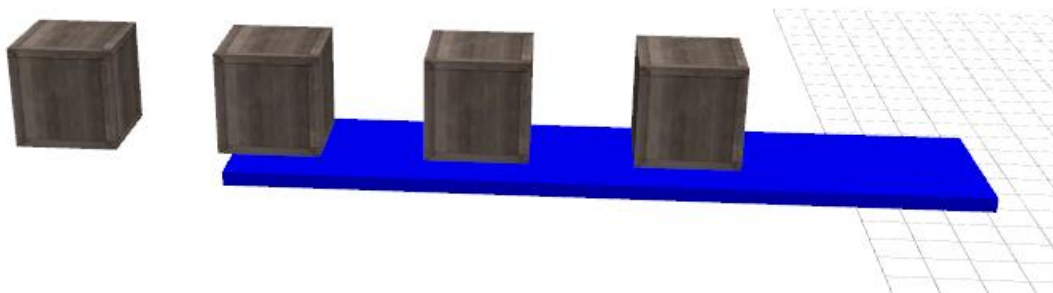
در این بخش به شرح قسمت های شبیه ساز می پردازیم. در زیر نمایی کلی از شبیه ساز را می توانید مشاهده کنید. با فعال کردن ساعت شبیه سازی (Start Timer) سیستم شروع به کار می کند. جعبه ها وارد سیستم می شوند و ربات آنها را بین ایستگاهها منتقل می کند.



## عناصر شبیه سازی

### ۱- ایستگاه ورودی

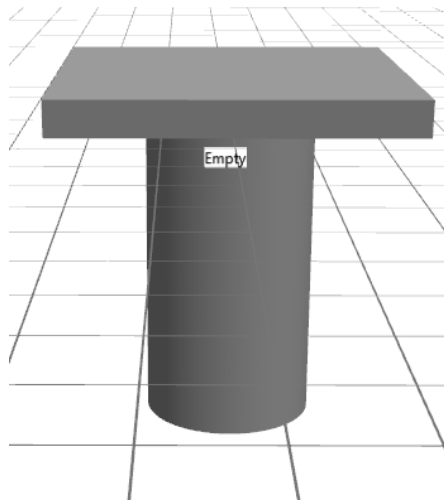
این ایستگاه را با صفحه ی آبی رنگی می توانید مشاهده کنید. جعبه ها از این قسمت وارد سیستم می شوند و در صورت مشغول بودن ربات یا پر بودن ایستگاه A وارد صفی در این ایستگاه می شوند.



ایستگاه ورودی

### ۲- ایستگاه های کاری

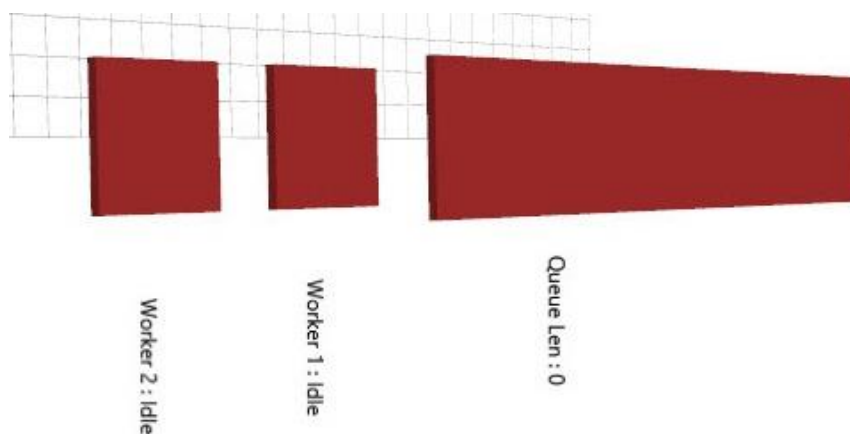
این ایستگاه را با رنگ خاکستری (و با شکلی خاص!) می توانید مشاهده کنید. در کنار هر یک از این ایستگاه ها یک نوشته ای وجود دارد که وضعیت ایستگاه را در هر لحظه نمایش می دهد.



یک ایستگاه کاری

### ۳- ایستگاه بازرسی

این ایستگاه شامل دو سکوی کاری و یک صف می باشد. از صفحات با رنگ قرمز برای نشان دادن آن استفاده شده است. در کنار هر کدام از دو سکوی کاری وضعیت فعلی کارگر مربوط به آن سکو، نشان داده شده است؛ همچنین در کنار صف انتظار، تعداد جعبه هایی که در حال انتظار هستند نمایش داده می شوند.



ایستگاه بازرسی



#### ۴- جعبه ها

جعبه ها عناصر اصلی شبیه سازی هستند. تمامی اطلاعات مربوط به زمان های ورود و خروج و شروع و اتمام سرویس، در هر جعبه به صورت مجزا ذخیره می شود. هر یک از جعبه ها به شکل مکعب مستطیل هستند. با پوسته (texture) به شکل جعبه چوبی! با کلیک کردن بر روی هر یک از جعبه ها، صفحه ایی مربوط به اطلاعات آن جعبه نشان داده می شود. اطلاعات مربوط به ورود و خروج، زمان های شروع خدمت دهی و پایان خدمت دهی.



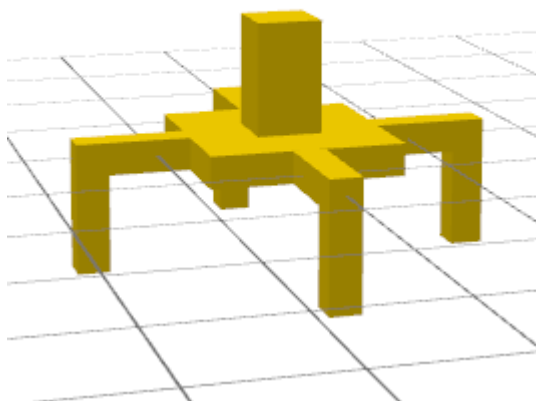
جعبه

BoxDetails	
Box Name	Box-3
Arrival Date	8:14:00
Departure Date	9:13:20
Service Start (A)	8:40:00
Service Start (B)	8:52:00
Service Start (C)	9:00:00
Service End (A)	8:41:00
Service End (B)	8:53:00
Service End (C)	9:01:00
Total Service Time	00:03:00
Total Wasted Time	00:56:20
<button>Close</button>	

نمایش دهنده اطلاعات مربوط به جعبه

## ۵- ربات سیستم

ربات سیستم را می‌توانید با رنگ زرد مشاهده کنید. قابل ذکر است که این ربات به طور کلی توسط کدهای برنامه نویسی ساخته شده است؛ و از هیچ نرم افزار سه بعدی- ساز در ساخت آن کمک گرفته نشده است.



ربات سیستم در وضعیت بی کار



ربات سیستم در حال بارگذاری جعبه

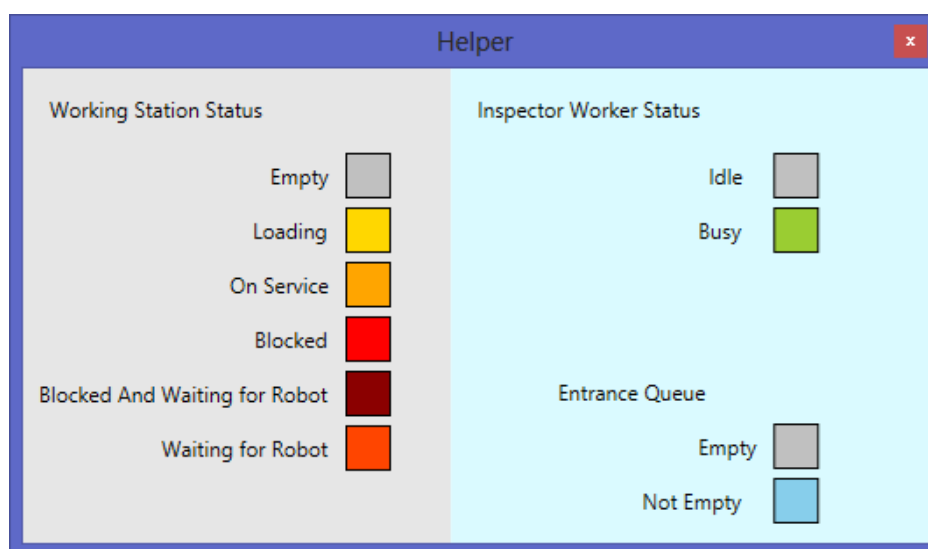
## ۶- نمایش دهنده وضعیت سیستم

در قسمت پایینی صفحه‌ی اصلی (پنجره) شبیه‌ساز، می‌توانید وضعیت فعلی سیستم را توسط این ابزارک مشاهده نمایید. این ابزارک به اختصاص دادن رنگ‌های مختلف به هر ایستگاه سیستم، وضعیت فعلی آن را نمایش می‌دهد.

برای فهمیدن معنی هر رنگ، کافیست از منو View گزینه‌ی Show Color Helper را انتخاب کنید.



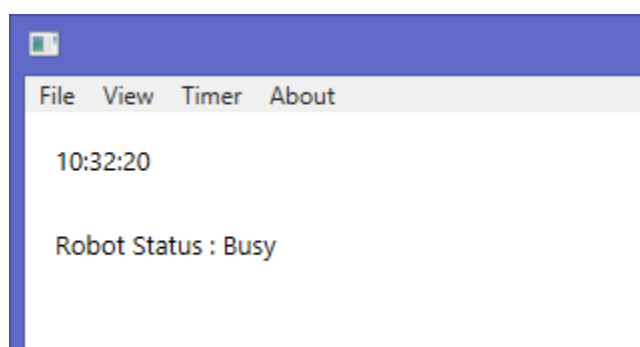
نمایش دهنده وضعیت سیستم



اطلاعات مربوط به معنی هر رنگ

## ۷- ساعت شبیه سازی و وضعیت ربات

در قسمت گوشه‌ی سمت چپ پنجره شبیه سازی، اطلاعات مربوط به زمان فعلی شبیه سازی و وضعیت فعلی ربات سیستم را می‌توانید مشاهده کنید.



## ۸- پیش‌آمد ها آتی

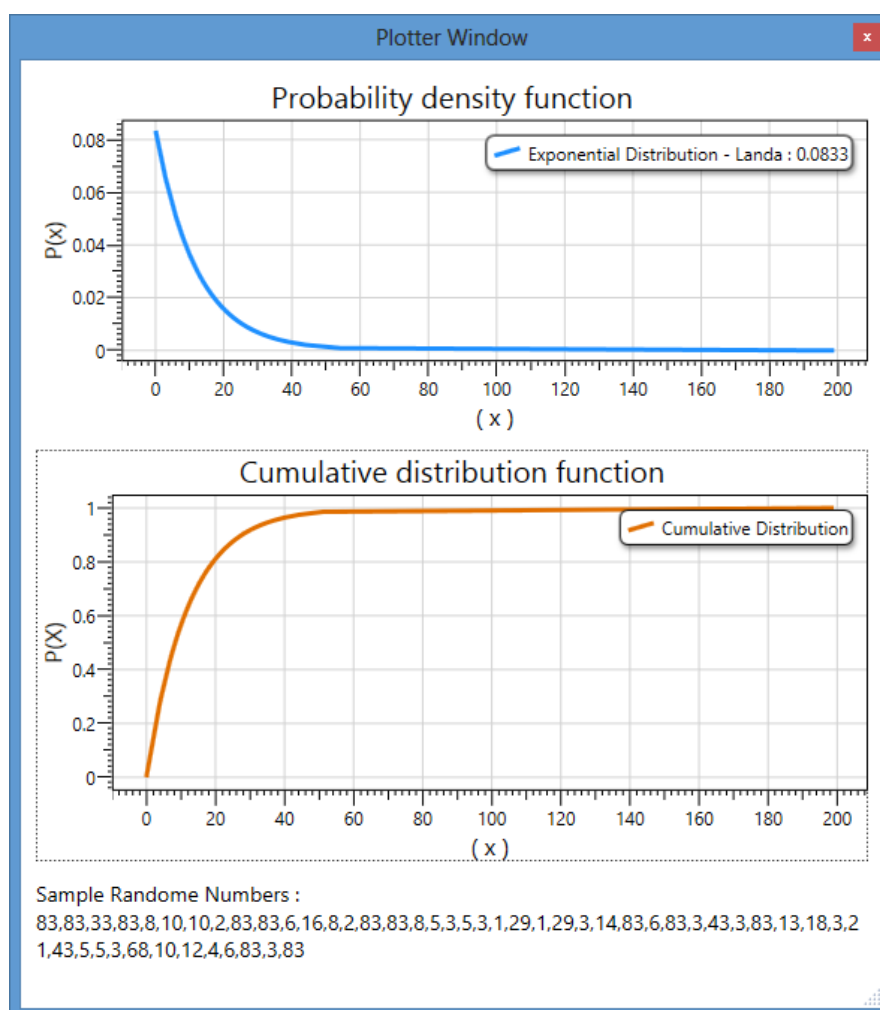
همراه با شبیه ساز، پنجره‌ای باز می‌شود که در آن تمامی پیش‌آمدهای آتی سیستم را می‌توانید مشاهده فرمایید.

## ۹- سایر عناصر

عناصری در صفحه می‌توانید مشاهده کنید که مربوط به موتور رندرینگ شبیه ساز است. اطلاعاتی در مورد تعداد فریم‌ها در ثانیه، زاویه دوربین، و سایر. برای غیر فعال کردن نمایش این موارد می‌توانید از منو View اقدام به حذف آن‌ها نمایید.

## ابزار مطالعه آماری برای توزیع نمایی با میانگین دلخواه

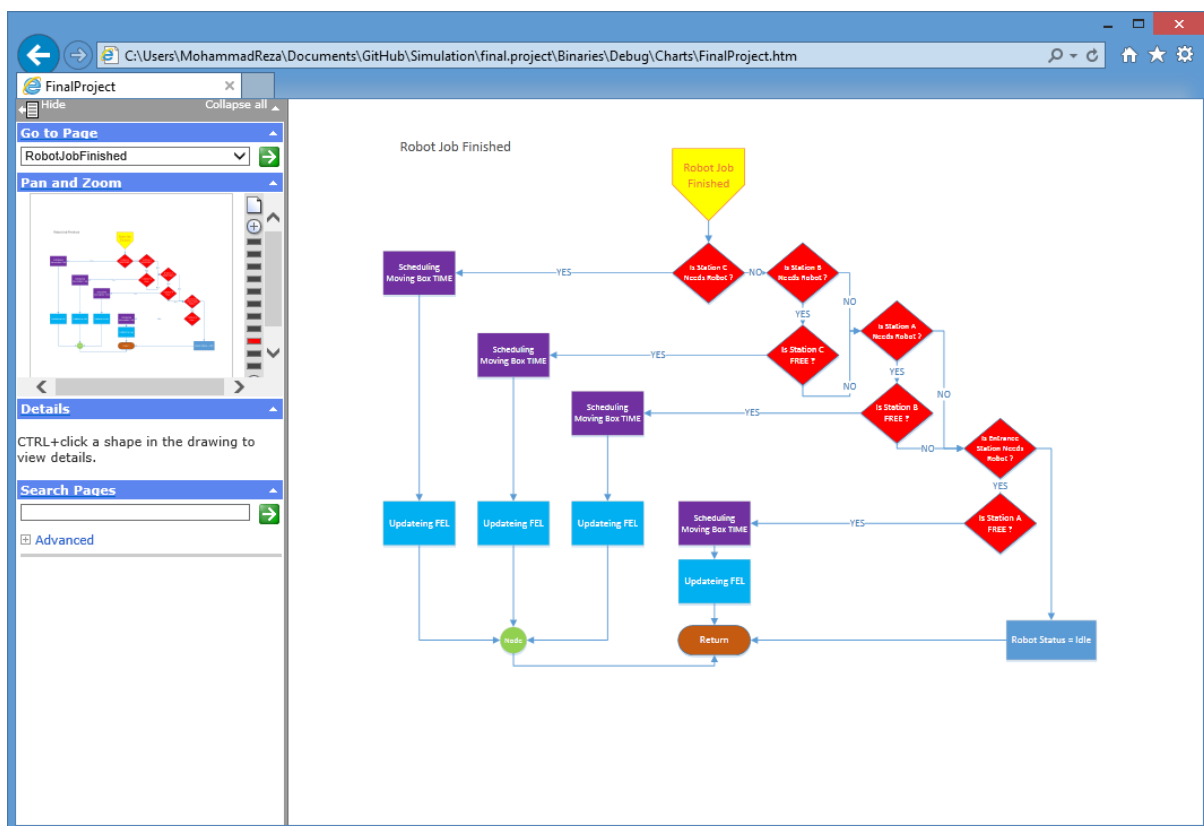
این قسمت گراف مربوط به تابع نمایی با میانگین دلخواه را رسم می‌نماید. در پایین دو نمودار ۵۰ عدد تصادفی، با توزیع نمایی (و میانگینی که کاربر وارد می‌کند) ساخته شده است.



Plotter

## نمودارهای جریان مربوط به شبیه سازی

در نرم افزار قابلیت مشاهده تمامی نمودارهای جریان فراهم آمده است. با کلیک بر روی گزینه Simulation Charts به صورت خودکار، مرورگر اینترنت اجرا می شود و در آن به راحتی امکان مشاهده نمودارهای جریان وجود دارد.



نمودارهای جریان

# Class Diagrams

