

پروژه سوم درس سیستم عامل

نیمسال دوم سال ۱۳۹۹-۱۴۰۰

شرح پروژه

قصد داریم در این پروژه زمان بندی بلادرنگ سیستم عامل را در یک رستوران شبیه سازی کنیم. فرض کنید رستوران تنها یک آشپز دارد و وظیفه ی آشپز این است که تمامی سفارشات را تا قبل از مهلت مشخص شده به دست مشتری رساند. سفارشات در این رستوران به گونه ای است که به صورت دوره ای تکرار می شوند و مهلت در یک دوره تعریف می شود.

لیست سفارشات

لیست سفارشات که ورودی برنامه است شامل اطلاعات زیر در مورد هر غذا است:

۱- زمان لازم برای پخت

۲- دوره تکرار سفارش

۳- مهلت تحویل سفارش (که برای هر غذا ثابت است)

سفارشات در اولین دوره به صورت همزمان دریافت می شود.

لیست سفارشات به صورت روزانه داده می شود. آشپز قبل از شروع به کار باید بررسی کند که

می تواند با توجه به لیست تمامی سفارشات را با توجه به دوره مشخص شده در هر سفارش به

دست مشتری رساند یا خیر. اگر نتواند باید پیغامی با عنوان "Can't handle orders" چاپ کند

و از برنامه خارج شود. (نتوانستن در پایان دوره به این معنا است که علاوه بر عبور از مهلت، در پایان دوره نیز یک سفارش آماده نشود.) همچنین در پایان روز لازم است آشپز مجموع زمان‌هایی که بیکار بوده است را اعلام کند. از آنجایی که جابه‌جا شدن بین سفارشات مشتری در هنگام پخت در کیفیت غذا تاثیر می‌گذارد برای ما مهم است تا آمار این جابجایی را نیز داشته باشیم بنابراین در پایان روز این تعداد جابجایی نیز توسط آشپز اعلام می‌شود.

اگر سفارشی از مهلت خود عبور کرد لازم است تا پیغام خطایی مانند زیر چاپ شود.

Food# deadline missed

اما در هر صورت آشپز پس از گذشت مهلت نیز سفارش را آماده می‌کند.

از آنجایی که پروژه شبیه‌سازی است، واحد زمان متغیری فرضی است که شما می‌توانید آن را با یک شمارنده در یک حلقه یا هر پیمایش حلقه مدیریت کنید.

برای بررسی عملکرد برنامه لازم است تا خروجی مشابه خروجی ذیل چاپ کنید که هر خط آن معادل سفارشی است که آشپز در حال پختن آن است.

0 Food1

1 Food1

2 Food2

3 Food3

4 Food3

5 Food3

6 Food2

7 Idle

Idle time = 1

Food1 waiting time = 0

Food2 waiting time = 4

Food3 waiting time = 3

فرمت فایل ورودی نیز هر خط به ترتیب شامل اسم غذا، زمان پخت، مهلت تحویل در هر دوره و

طول زمان دوره است:

	burst	deadline	period
Food1	2	2	8
Food2	1	3	4
Food3	3	6	8
...			

به عنوان مثال خط اول به این معناست که غذای اول دو واحد زمانی برای پخت نیاز دارد و هر ۸

واحد زمانی یکبار این سفارش تکرار می شود و در هر دوره تا دو واحد اول زمانی آن دوره مهلت

دارد. لازم به ذکر است برای این که بدانیم در هر روز سفارشات چند بار تکرار می شوند بین

دوره های غذا ک.م.م میگیریم عدد بدست آمده کل زمانی است که آشپز در رستوران حضور دارد.

در مثال بالا این عدد ۸ است.

در هنگامی که دو غذا(فرآیند) اولویت یکسانی داشته باشند در صورتیکه یکی از غذاها در حال

پخته شدن باشد، آشپز به پختن آن ادامه می دهد و در صورتی که هیچکدام در حال پخته شدن

نباشند غذایی آماده می شود که زمان پخت کمتری دارد.

برای شبیه سازی زمانبندی لازم است تا الگوریتم های زیر را پیاده سازی کنید.

1. اول زودترین مهلت (Earliest Deadline First)

2. نرخ یکنواخت (Rate Monotonic)

3. اول کمترین لختی (Least Laxity First)

a. در این الگوریتم در صورتی که سفارش، قبل از مهلت آن آماده نمی‌شود لازم است

هر زمانی که برای آشپز قابل پیش‌بینی بود عبارت زیر چاپ شود. عدد اول نشان

دهنده زمان مشخص شدن آن است:

Food# will miss deadline

نکات مهم

۱- محدودیتی در زبان پیاده‌سازی پروژه وجود ندارد.

۲- پروژه به صورت انفرادی یا در گروه‌های دو نفره قابل انجام است.

۳- در صورت مشاهده هرگونه شباهت بین گروه‌ها و اثبات تقلب نمره ۱۰۰- به هر دو گروه داده می‌شود.

۴- خروجی خواسته شده باید برای هر الگوریتم به صورت جداگانه چاپ شود و زمان‌های بیکاری آشپز و

مجموع زمان انتظار هر سفارش در الگوریتم‌ها را با یکدیگر مقایسه شوند.

۵- موارد تحویلی شامل سورس کد شما به همراه مستندات پروژه است. مستندات باید شامل توضیحاتی در مورد

هر تابع پیاده‌سازی شده باشد.

۶- فایل‌های نهایی در قالب زیر ارسال شود :

FirstName_LastName_StudentNumber_PR3.zip

موفق باشید

تیم حل پروژه سیستم عامل