

(به نام خداوند بخشندهی مهربان)



درس برنامه نویسی پیشرفته

تمرین سری سوم

دانشکدهی مهندسی کامپیوتر

دانشگاه علم و صنعت ایران

استاد مرضیه ملکی مجد

نیم سال دوم ۱۴۰۰-۱۳۹۹

مهلت ارسال: ۱۳۹۹/۱/۲۴

مبحث تمرینات:

مباحث گذشته و enum مدیریت خطا

نکات کلی:

* برای هر ۴ سوال این تمرین، مدیریت خطا باید به صورت کامل اجرا شود و برنامه‌ی شما تحت هیچ شرایطی برنامه به علت خطای زمان اجرا متوقف نشود. از جمله مواردی که ممکن است باعث توقف برنامه شما شود، وجود نداشتن فایلی با نام مشخص شده یا تفاوت فرمت ورودی توسط کاربر با فرمت مورد نظر شما یا مشکل داشتن فایلی که اطلاعات ورودی در آن قرار دارد و ... است. تمامی این موارد باید توسط شما در نظر گرفته شود و از متوقف شدن برنامه هنگام اجرا توسط مدیریت خطا جلوگیری کنید.

* در سوال یک تا سه شما باید از `enum` استفاده کنید. در این موارد اگر در هر بخش شما به جای `enum` از رشته استفاده کنید یا از `cast` کردن‌های بی مورد استفاده کنید، نمره‌ای از آن بخش به شما تعلق نخواهد گرفت. لذا دقت لازم را به عمل آورید.

Search Engine. ۱

هدف این است که نرم افزاری برای یک مرجع اطلاعات کامپیوتری طراحی کنیم. کاربران این شرکت از نرم افزار، برای سرچ کردن راجب سرفصل های مرتبط به حوزه ی کامپیوتر استفاده میکنند.

حال این شرکت قصد دارد از بین سرچ کاربران خود ۶ مورد خاص را مورد بررسی قرار دهد و ببیند هرکدام از این ۶ مورد چند بار مورد جستجو قرار گرفته است..

مواردی که شرکت زیر نظر دارد Web(83) ,AI(36) ,IOT(45) ,IT(21), Database(55) ,ImageProcessing(27) این موارد هستند که هر کدام کد مخصوص به خود را دارند که در نتیجه ی آخر باید هر مورد به همراه کد آن چاپ شود.

برنامه دارای دو بخش Admin و User است.

:Admin

در این بخش برای ورود نیازمند وارد کردن کلمه عبور است بدین صورت که اگر در میان فایل های برنامه فایلی به نام Password.txt وجود نداشت رمز ورود "Hello@P" است اما اگر فایل رمز ورود وجود داشت باید رمز ورودی کاربر با رمز موجود داخل فایل همخوانی داشته باشد.

*بخش امتیازی ۱: رمز داخل فایل Password.txt به وسیله ی الگوریتمی دلخواه رمزنگاری بشود که به شکل واضح قابل مشاهده نباشد.(روش های متفاوتی برای رمزنگاری وجود دارد. از جمله ساده ترین آن ها می توان به cipher encoding اشاره کرد و در موارد پیچیده تر شامل رمزنگاری RSA و ... می شود).

در این بخش منویی از گزینه های Count ,ChangePassword ,Exit وجود دارد که تا زمانی که گزینه Exit انتخاب نشده، نشان داده میشود و وقتی این گزینه انتخاب باید از بخش Admin خارج شود.

وقتی مدیر روی گزینه Count کلیک میکند باید تعداد جستجوی هر کدام از موارد به صورت لیست چاپ بشود بدین صورت که ابتدا کد آن مورد سپس نام و بعد تعداد سرچ های آن مورد در یک سطر چاپ شود.

وقتی روی گزینه ChangePassword کلیک بشود باید ابتدا رمز فعلی وارد شود سپس رمز جدید به همراه تکرار آن وارد شود و داخل Password.txt ذخیره شود.

User:

این بخش شامل یک Search Bar است که همواره در حال جستجوی سرفصل ورودی توسط کاربر است تا جایی کاربر کلمه ی "exit" وارد کند و وقتی کلمه خروج را وارد کرد از بخش User خارج بشود.

*نکته بسیار بسیار مهم:

تمامی ورودی های کاربر در برنامه شما باید به صورت enum نگهداری و بررسی شوند و پس از اینکه کاربر در ورودی کلمه ای را وارد کرد، در طول برنامه شما نباید به صورت string در نظر گرفته شود و باید به صورت enum از آن استفاده کنید.

ورودی کاربر به شکل های مختلف باید بتواند دریافت بشود برای مثال کاربر میتواند داخل search bar کلمه ی مورد سرچ اش WEB یا Web یا weB یا web باشد و همه قابل قبول است.

مثال:

Search Bar: web

Search Bar: Web

Search Bar: WEB

Search Bar: weB

تمامی موارد ورودی بالا صحیح می باشد و به عنوان یکی از اعضا enum ساخته شده ینی Web با کد ۸۳ شناخته میشود و یکی به تعداد سرچ این مورد اضافه کند و در حد یک خط راجب این موضوع توضیحاتی چاپ کند.(این توضیحات راجب هر موضوع را در اینترنت به دلخواه سرچ کنید و بنویسید.)

اگر ورودی کاربر شامل آن موارد مورد نظر شرکت نبود exception ایجاد میشود شما نیز باید از این خطا را مدیریت کنید و به کاربر اعلام کنید که این مورد در آینده افزوده میشود(منظور این است که اگر ورودی که توسط کاربر وارد می شود جزو شش مورد بالا نبود، اعلام کنید که این مورد در آینده موجود می شود و در این حالت نباید برنامه ی شما متوقف شود و از کاربر دوباره ورودی بگیرید.) و اگر ورودی کاربر از موارد مورد نظر شرکت بود مطالبی در حد یک خط راجب این موضوع چاپ کند. (این مطالب را از اینترنت سرچ بکنید و یک خط به عنوان توضیحات چاپ بکند.)

در بخش Admin زمانی که در حال تغییر رمز هستید وقتی از File استفاده میکنید از خطا هایی احتمالی جلوگیری شود بدین صورت که چه خطا وجود داشته باید چه خطا بوجود نیاید برنامه ی شما باید در پایان کار با File رفرنس آن را پایان دهد.(راهنمایی: از finally برای اجرای بخش آخر استفاده کنید.)

***بخش امتیازی ۲:** سعی کنید برنامه شما دارای ظاهر زیبایی باشد. برای این مورد می توانید از رنگ بندی متفاوت متن و زمینه ی کنسول استفاده کنید.

۲. باغ وحش

مسئول یک باغ وحش می‌خواهد برای نگهداری حیوانات از برنامه مدیریت اطلاعات استفاده کند. به همین منظور نیاز هست کلاسی برای مدل کردن اطلاعات حیوانات بنویسد. برای هر حیوان یک شی از این کلاس ساخته خواهد شد. حیوانات زیر در باغ وحش وجود دارند

شیر (Lion)، ببر (Tiger)، خرس (Bear)، میمون (Monkey)، فیل (Elephant)، زرافه (Giraffe)، جغد (Owl).

شما باید کلاسی به نام Care برای نگهداری اطلاعات حیوانات این باغ وحش طراحی کنید. اطلاعات هر حیوان شامل کد حیوان (ID)، نام حیوان (name)، محل نگهداری (location)، نوع غذایی که می‌خورد (food)، تعداد حیوان (number) و ساعت غذا دادن در روزهای هفته است. این اطلاعات فیلدهای کلاس Care را تشکیل می‌دهند. ویژگی کد حیوان و نام حیوان نباید خارج از کلاس در دسترس باشند. نام هر حیوان باید مقداری از یک نوع داده ای enum باشند که خودتان تعریف کرده اید.

نکته ۱: دقت کنید که عدد متناظر با نام حیوان در enum باید همان کد حیوان باشد.

نکته ۲: دقت کنید که تعداد وعده های غذایی سه وعده میباشد. پس schedule باید به صورت آرایه ی سه تایی تعریف شود.

نکته ۳: جایی از کد برنامه اسم حیوانات را به صورت رشته نگهداری نکنید و فقط از enum ای که تعریف کرده اید استفاده کنید.

کدهای مربوط به هر حیوان به صورت زیر است.

شیر = ۹۶۰۰ ، ببر = ۹۶۱۱ ، خرس = ۹۶۲۲ ، میمون = ۹۶۳۳ ، فیل = ۹۶۴۴ ، زرافه = ۹۶۵۵ ، جغد = ۹۶۶۶

کلاس Care باید موارد زیر را داشته باشد:

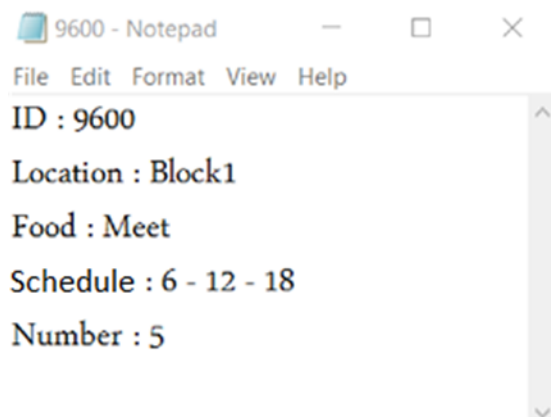
سازنده کلاس باید به عنوان پارامتر ورودی کد حیوان (ID)، نام حیوان (name)، محل نگهداری (location)، غذایی که میخورد (food)، تعداد حیوان (number) و ساعت غذا دادن در روزهای هفته (schedule) را بگیرد.

متد SetSchedule با توجه به نوع غذایی که هر حیوان مصرف میکند، ساعت غذا دادن به آن را به صورت زیر محاسبه کند و در فایل مربوط به حیوان نیز تغییر ایجاد کند. (این متد ساعت غذا را عوض می کند)

اگر حیوان گیاه‌خوار بود، ابتدا یک عدد را به صورت رندوم بین ۶ تا ۲۲ قرار دهد. سپس دو عدد دیگر را به طوری که فاصله ی بین هر دو وعده کمتر از ۲ ساعت و بیشتر از ۴ ساعت نشود انتخاب کنید و این سه عدد را به عنوان ساعات وعده های غذایی جدید ذخیره کنید.

اگر حیوان گوشت‌خوار بود، فایل مربوط به آن حیوان را باز کند و در صورتی که ساعت وعده آخر غذایی حیوان برابر با ۲۲ بود آن را با عدد رندومی بین ۱۷ تا ۲۱ جایگزین کند.

متد SaveToFile که اطلاعات حیوانات را در فایل ذخیره کند. برای ذخیره هر حیوان باید فایلی با نام کد حیوان ساخته شود و اطلاعات هر حیوان سطر به سطر مشابه تصویر زیر ذخیره شوند.



```
9600 - Notepad
File Edit Format View Help
ID : 9600
Location : Block1
Food : Meet
Schedule : 6 - 12 - 18
Number : 5
```

متد AllInfo که به صورت استاتیک تعریف می شود و به عنوان ورودی آرایه ای از اشیای کلاس را می گیرد و یک فایل در خروجی تولید می کند که در هر خط آن اسم حیوان و آی دی آن آمده است.

در قسمت main این سوال:

اول دقت کنید که باید پیغام مناسب برای راهنمایی کاربر هنگام دریافت ورودی و نمایش خروجی چاپ کنید.

دوم که ترتیب کارهایی که انجام می دهد به این صورت است:

ابتدا باید اطلاعات حیوانات از ورودی دریافت کنید. نحوه دریافت به این صورت است که در خط اول تعداد حیواناتی که می‌خواهید اطلاعات نگهداری آن‌ها را ذخیره کنید از کاربر دریافت میکنید. پس از آن برای هر حیوان اطلاعات آن حیوان را از کاربر دریافت می کنید. (برای مثال به شکل دقت کنید).

```
Enter number of animals to save
1
Enter the code
9600
Enter the location
Block1
Enter the food
Meet
Enter the schedule
6 - 12 - 18
Enter the number
5
```

پس از دریافت اطلاعات هر حیوان، شی متناظر از این حیوان ساخته می شود. اطلاعات حیوان در فایل ذخیره می شود (از متد Save to file که نوشته اید باید استفاده کنید). اشیا ساخته شده باید در آرایه ای از نوع Care نگهداری شوند.

پس از دریافت همه ی اطلاعات حیوانات، شما آرایه ای از اشیا کلاس Care دارید. با استفاده از متد SetSchedule باید ساعت غذادهی هر حیوان را تغییر دهید و فایل مربوط به حیوان را آپدیت کنید. در انتها باید متد AllInfo را بر روی اشیایی که ساخته اید صدا بزنید.

۳. سامانه نمره دهی دانشجویان بر اساس مدت زمان حضور

امروزه و باتوجه به مجازی شدن دانشگاه‌ها، مدت زمان حضور دانشجویان در سامانه‌های کلاس‌های مجازی، اهمیت زیادی پیدا کرده است. گرچه سامانه‌های کنونی ساعت حضور دانشجویان را ثبت می‌کنند، اما با این حال تبدیل این ساعات حضور به نمره ی مورد نظر استادان هنوز هم زمان زیادی می‌گیرد. بنابراین تصمیم گرفته شده تا سامانه‌ای طراحی شود تا با وارد کردن ساعت حضور، نمره ی فرد را هم خروجی بدهد.

شما باید یک کلاس پیاده سازی کنید که به ازای هر دانشجو، یک شی از این کلاس داشته باشید. شما باید برنامه ای بنویسید که اطلاعات دانشجویان را دریافت کند و به ازای هر دانشجو یک شی بسازد. به این ترتیب آرایه ای از اشیا خواهید داشت که ادامه برنامه با استفاده از آن ها خواهد بود. توضیحات کلاس، نحوه دریافت اطلاعات برای ساخت شی و خروجی موردنظر برنامه در زیر توضیح داده شده است.

کلاس دانشجو:

شامل فیلدهایی برای نگهداری نام، نام خانوادگی، شماره دانشجویی و تعداد ساعات حضور دانشجو در کلاس و نوع دانشجو است. از آنجایی که این سامانه در ابتدا فقط قرار است برای درس‌هایی که یک جلسه در هفته تشکیل می‌شوند ارزیابی شود، حداکثر ساعت حضور در کلاس ۸ ساعت است. لازم به ذکر است که به علت اینکه ابتدا لازم است این برنامه به صورت آزمایشی منتشر شود، فعلا فقط برای یکی از کلاس‌ها فعال می‌شود و فرض بر این است که هر دانشجو فقط یک کلاس را شرکت می‌کند.

دانشجویان در دو دسته ی ابتدایی دستیاران آموزشی (TA)، دانشجویان عادی (Normal) قرار می‌گیرند. همچنین پس از توطئه ی عده ای از دانشجویان بر علیه دستیاران آموزشی در کلاس‌های دیگر (مانند زبان تخصصی) دسته ی جدیدی به نام شورش (Rebel) به این دسته بندی اضافه

شده است. بنابراین در نهایت، ما سه نوع دانشجو داریم. دانشجویان عادی (Normal)، دستیاران آموزشی (TA) و دانشجویان شورشی (Rebel)

نوع دانشجو باید با نوع داده ای enum پیاده سازی شود. بنابراین ذخیره کردن و پیاده سازی این مورد به صورت رشته string و یا استفاده از cast کردن بی مورد و اضافی، موجب میشود که از این بخش نمره ای به شما تعلق نگیرد.

این کلاس باید سازنده ای با پارامترهای مورد نیاز برای مقداردهی همه فیلدها داشته باشد در حالی که اگر تعداد ساعات حضور دانشجویی وارد نشود به صورت پیشفرض مقدار ۶ ساعت برای آن دانشجو در نظر گرفته بشود.

شماره دانشجویی باید ۸ رقم باشد و دو رقم ابتدایی آن باید ۹۵ یا ۹۶ یا ۹۷ یا ۹۸ یا ۹۹ باشد.

متمدی برای محاسبه نمره‌ی دانشجو داشته باشد. در این متد بسته به شرایط دانشجویان، نمره‌ی دانشجو محاسبه می‌شود و به عنوان خروجی برگردانده می‌شود. نحوه نمره دادن در زیر توضیح داده شده است:

دانشجویان عادی بسته به ساعت حضور خود نمره می‌گیرند. اگر ساعت حضور آنها کمتر از ۳ ساعت باشد، نمره ای دریافت نمی‌کنند. اگر بین ۴ تا ۶ ساعت حضور داشته باشند، ۶ نمره و اگر بین ۶ تا ۸ ساعت حضور داشته باشند، ۸ نمره می‌گیرند.

دستیاران آموزشی، یک و نیم برابر ساعت حضور خود نمره می‌گیرند.

دانشجویان شورشی هم به اندازه‌ی نصف زمان حضور خود نمره می‌گیرند.

دقت داشته باشید که شماره دانشجویی هر دانشجو منحصر به فرد است و در صورت تکراری بودن آن، اطلاعات نباید ثبت شود و از کاربرد درخواست شود تا اطلاعات را دوباره وارد کند.

Main:

در خط اول برنامه، تعداد دانشجویان داده می‌شود. در n خط بعدی اطلاعات دانشجویان داده می‌شود. با استفاده از کلاس گفته شده باید برای هر یک از دانشجویان یک شی ساخته شود. سپس در خروجی کنسول، به ترتیب در n خط نام و نام خانوادگی و نمره ی افراد چاپ شود. نمونه ای ورودی و خروجی برنامه در ادامه برای شما آورده شده است.

ورودی:

Enter number of people:

4

Enter name, family, code, hours and group of person:

Morteza,Shahrabi, 99888877,TA

Enter name, family, code, hours and group of person:

Parsa,Dorali, 99888877,TA

Invalid code. Please enter information again.

Enter name, family, code, hours and group of person:

Parsa,Dorali, 99884466,8,TA

Enter name, family, code, hours and group of person:

Hesam,Hosseini,98882155,5,Normal

Enter name, family, code, hours and group of person:

Mahmoud,Mahmoudi, 98542211,8,Rebel

خروجی:

Morteza Shahrabi: 12

Parsa Dorali: 16


Hesam Hosseini: 6

Mahmoud Mahmoudi: 4

دقت کنید که تمامی خطاهای احتمالی در طول برنامه از جمله وارد کردن عددی بزرگتر از ۸ یا کمتر از صفر به عنوان ساعت حضور در کلاس، یا وارد کردن ورودی‌های نامعتبر برای شماره دانشجویی و گروه دانشجو و سایر موارد احتمالی باید توسط شما در نظر گرفته شود و پیغام مناسب برای آن نمایش داده شود.


۴. ستاره چاپ کن

هدف نوشتن برنامه ای است که اطلاعات یک فایل متنی را با یک سری تغییرات به یک فایل جدید منتقل کند. به این صورت که در فایل جدید تمامی فضاهاى خالی بین کلمات تبدیل به * شده اند. همچنین باید در خروجی کنسول تعداد خطوط، تعداد اعداد، تعداد ستاره ها و تعداد حروف صدادار فایل نمایش داده شود. به نمونه ورودی و خروجی زیر دقت کنید.

 a1 - Notepad

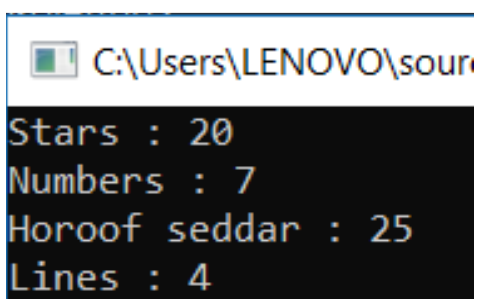
```
File Edit Format View Help
hi my name is john. how are you ?
I am student. I see you at 13 : 00.
good bye !
405 error|
```

در فایل خروجی، باید محتوای زیر را ببینیم.

 a2 - Notepad

```
File Edit Format View Help
hi*my*name*is*john.*how*are*you*?
I*am*student.*I*see*you*at*13*:*00.
good*bye*!
405*error|
```

و در کنسول باید خروجی مطابق شکل زیر ببینیم.



```
C:\Users\LENOVO\source
Stars : 20
Numbers : 7
Horroof seddar : 25
Lines : 4
```

نام فایل‌های ورودی و خروجی باید در ابتدای برنامه از طریق ورودی کنسول از کاربر گرفته شود.