

به نام خدا



درس برنامه نویسی پیشرفته

تمرین هفتم

دانشکده مهندسی کامپیوتر

دانشگاه علم و صنعت ایران

استاد مرضیه ملکی مجد

نیم سال دوم ۱۴۰۰-۱۴۰۱

مهلت ارسال :

۱۴۰۱ / ۳ / ۱۴

مبحث:

جنریک - لینک

مسئول تمارین :

آریا شهنسوار

فهرست

- آداب نامه تمرینات ۳
- نکات تمرین سری هفتم ۴
- تمرین ۱. کریستوفر رابین ۵
- تمرین ۲. IMDB ۷
- تمرین ۳. استک ۸
- تمرین ۴. شطرنج (امتیازی) ۱۲

آداب نامه تمرینات

- پاسخ تمامی سوالات تنها به زبان C# قابل قبول می باشد
- علیرغم اعتماد کامل تیم تی ای به شما دانشجویان عزیز ، تمامی کدهای شما با سایر دانشجویان بصورت خودکار و توسط برنامه مقایسه خواهند شد . همچنین در طول ترم ، از تمامی پاسخ های شما ارائه گرفته خواهد شد و نحوه کار تمامی بخش های هر سوال از شما پرسیده خواهد شد ، لذا از کپی نمودن کد دوستانتان خودداری کنید و تمامی پاسخ ها ، کد خودتان باشد . همچنین از آنجایی که مشورت و هم فکری با سایر دوستان بسیار کار پسندیده و مفیدی است - برخلاف کپی کردن کد (:-) در صورت هم فکری با دانشجوی (دانشجویان) ، نام وی را بصورت کامنت شده در ابتدای کد خود بنویسید .
- برای ارسال تمارین در طول ترم ، در مجموع ۱۰ روز می توانید تاخیر داشته باشید و در صورتی که جمع تاخیر دانشجویی بیشتر از ۱۰ روز شود ، تمرین وی قابل قبول نخواهد بود لذا تلاش کنید تمرینات را در زمان مقرر در سامانه آپلود کنید
- برای هر تمرین در زمان ددلاین دانشجو باید درخواست خود را مبنی بر ارسال با تاخیر به تی ای ها اعلام نماید
- در تمامی تمرینات سعی شده است که سوالات ساده تر در ابتدا و سوالات دشوار تر در انتهای فایل قرار گیرند (از ساده به دشوار مرتب شده اند)
- در صورت وجود هرگونه سوال در مورد تمرینات ، سعی کنید تا جایی که امکان دارد سوال خود را در گروه پرسید چرا که شاید سوال شما ، سوال دوستان نیز باشد و دوستانتان نیز بتوانند از پاسخ سوال شما بهره ببرند .

نکات تمرین سری هفتم

- سوالات را در سامانه کوئرا و در قسمت تمرین سری هفتم آپلود نمایید .
- در تمامی سوالات باید مدیریت خطا بطور کامل و دقیق صورت گیرد و بخشی از نمره هر سوال به exception handling درست و مناسب تعلق می گیرد .
- در پیاده سازی کلاس ها ، دقت کنید که تمام مواردی که در سوال از شما خواسته شده است با دقت پیاده سازی شده باشد .
- فایل های مورد نیاز سوال دوم در کوئرا و کانال درس قرار داده شده است .
- سوال چهارم تمرین، امتیازی بوده و نمره آن منوط به پیاده سازی مفاهیم پیشرفته درس می باشد
- از آنجایی که هر سوال توسط یک تی ای طرح شده است ، تنها تی ای طراح آن سوال می تواند شما را بصورت دقیق راهنمایی کند به همین منظور طراح هر سوال در زیر نوشته شده است تا در صورت ابهام و پرسش در مورد هر سوال ، در صورتی که نیاز به پرسش سوال بصورت انفرادی در پیوی هست ، به تی ای مربوطه مراجعه بفرمایید
- سوال ۱ : خانم ریحانه شاهرخیان
- سوال ۲ : خانم ریحانه شاهرخیان
- سوال ۳ : آقای کامیار مرادیان
- سوال ۴ : آقای محمد علی صمدی - آقای آریا شهنسوار

تمرین ۱ . کریستوفر رابین

کریستوفر رابین به جنگل بازگشته است و قصد دارد دوستانش را ملاقات کند . بنابراین تصمیم میگیرد قرار ملاقات را در کلبه کوچکش بگذارد . به دلیل اینکه کلبه اش کوچک بوده و به اندازه یک نفر ظرفیت دارد، دوستانش باید به ترتیب با او ملاقات کنند . کریستوفر رابین برای ملاقات با دوستانش امتیازهایی به آنها داده است . اما با توجه به اینکه دوستانش شیطننت هایی دارند لذا هر کدام تلاش میکنند تا زودتر با او ملاقات کنند . کریستوفر رابین اعلام کرده است به دلیل مشغله کاری اش، امروز میتواند ۳ نفر از دوستانش را ببیند.

۱. **پو:** یک خرس زرد رنگ مهربان که عاشق عسل است . امتیاز ۵ برای او در نظر گرفته شده است . در واقع بالاترین امتیاز ثبت شده برای اوست . پس او همیشه میتواند اولین نفر کریس توفر را ملاقات کند.

۲. **پیگنت:** خوک صورتی بامزه ای که کمی ترسو است . امتیاز ۴ برای او در نظر گرفته شده است.

۳. **تایگر:** یک ببر شیطون و همیشه خندان است . امتیاز ۳ برای او در نظر گرفته شده است.

۴. **روو:** بچه کانگرو شیطون که رفیق تیگر هست . امتیاز ۲ برای او در نظر گرفته شده است.

۵. **ایور:** خر خسته که همیشه غرغر میکند . امتیاز ۱ برای او در نظر گرفته شده است

پیاده سازی : در این سوال میخواهیم با کلاس generic بیشتر آشنا شوید . در این سوال شما باید یک رابط و شش کلاس پیاده سازی کنید :

رابط IPersonality :

شامل دو ویژگی (property) ، نام از نوع string و امتیاز از نوع int است، همچنین دارای یک متد به نام Personality بدون پارامتر ورودی و خروجی ای از نوع string است .

کلاس ها :

دوستان کریستوفر رابین به صورت کلاس تعریف میشوند. اسامی کلاسها باید بر اساس نوع حیوانی که هستند باشد. مثلا برای پو کلاسی با نام Bear ، برای تیگر کلاسی با نام Tiger و غیره داریم. در همه کلاسها لازم است رابط IPersonality را پیاده سازی کنید. متد Personality باید تمامی تعاریفی که از دوستان کریستوفر در زیر به رنگ آبی مشخص شده را به عنوان رشته برگرداند.

مثلا برای کلاس پو خواهیم داشت : پو خرس زرد رنگ، عاشق عسل، امتیاز ۵

Pooh is yellow bear.

He loves honey.

His name's rate is 5 => from property

توجه ۱ : دقت کنید باید برای این قسمت ۵ کلاس بنویسید

کلاس Friend :

این کلاس به صورت generic تعریف شود و تایپ های مورد قبول آن ، تایپ هایی هستند که رابط IPersonality را پیاده سازی کرده باشند. تایپ جنریک کلاس را T نام گذاری کنید . در این کلاس یک فیلد از نوع T وجود دارد. سازنده ای با پارامتر ورودی از نوع T نیز پس از پیاده سازی کلاسها، در Main برنامه خواهیم داشت:

```
Friend<Bear> bear = new Bear ("Pooh", 5)
```

```
Friend<Tiger> tiger = new Tiger ("Tiger",3)
```

... (و برای سایر دوستانش به همین ترتیب خواهیم داشت)

خروجی متد موجود در کلاس Friend را برای هر یک از شی های ساخته شده چاپ کنید. و در نهایت یک متد تعریف کنید که به عنوان خروجی ، همان خروجی متد Personality را برمیگرداند.

تمرین ۲ . IMDB

موضوع تمرین پردازش برخی از داده های مربوط به فیلم های IMDB است .
فایل IMDB-Movie-Data.csv شامل داده هایی مربوط به فیلم ها در سایت IMDB است .
پردازش اولیه در فایل CSVParser.cs که ضمیمه شده است، پیاده سازی شده است . کاری که شما باید انجام دهید این است که Extension Method های مناسبی را پیاده سازی کنید و از آنها برای پاسخگویی به سوالاتی که در زیر مطرح شده است ، استفاده کنید.

- ۱ . ژانر فیلم هایی که مدت زمان آنها زیر ۱۰۰ دقیقه است را چاپ کند.
- ۲ . نام کارگردان فیلم هایی که وین دیزل در آن بازی کرده است.
- ۳ . مشخصات فیلمی با بیشترین رای در سال ۲۰۱۶ را برگرداند.
- ۴ . نام فیلم ها به کارگردانی Bryan Singer را همراه با میزان فروش آنها به ترتیب نزولی چاپ کند.
- ۵ . مجموع فروش تمامی فیلمهای اکران شده در سال ۲۰۱۱ را چاپ کند.
- ۶ . میانگین فروش تمامی فیلمهای اکران شده در سال ۲۰۱۴ را چاپ کند.
- ۷ . ۱۰ تا از پرفروش ترین فیلم های ژانر اکشن که طولانی تر از دو ساعت هستند.
- ۸ . فیلم هایی که در نامشان عدد وجود دارد.
- ۹ . فیلم های Anne Hathaway, Jennifer Lawrence به ترتیب قدیم به جدید و بیشترین rating به کمترین . (فیلم های هر بازیگر نه اینکه هر دو با هم)
- ۱۰ . مقایسه تعداد فیلم های درام و کمدی دارای rating بیشتر از ۸ .
- ۱۱ . نام بازیگری را بیابید که تعداد فیلم های بدی که بازی کرده است از بقیه بازیگر ها بیشتر است (فیلم بد را به معنی rating کمتر از 7 در نظر بگیرید)
- ۱۲ . نام فیلم یا فیلم هایی که تعداد حروفشان از Prometheus بیشتر است را برگرداند.
- ۱۳ . نام فیلم هایی که ژانر آنها Action است یا سال ساخت آنها ۲۰۱۴ است.
- توجه ۱ :** در این مورد اجازه ی استفاده از || را در where ندارید پس متد جدیدی را پیدا کنید (
- ۱۴ . نام ۱۰ فیلم برتر ژانر Comedy بجز ۳ فیلم برتر اول را برگرداند.
(درواقع از ۱۴امین فیلم تا ۱۰ام فیلم برتر)

تمرین ۳ . استک

تاکنون با دو نوع داده آرایه و لیست آشنا شده اید. همانطور که می دانید در هر دوی آن ها به تمام اعضای موجود ، دسترسی دارید. اما گاهی اوقات ، تنها به اعضای ابتدایی و انتهایی داده نیاز خواهیم داشت . در چنین موقعیتی بهتر است که از نوع داده ای متفاوتی استفاده کنیم.

در این سوال از شما می خواهیم که یک داده ساختار با عنوان پشته (Stack) که در ادامه با آن ها آشنا می شوید را پیاده سازی کنید. موارد لازم جهت پیاده سازی این نوع داده در ادامه آورده شده است.

رابط IData :

این اینترفیس جنریک می باشد و قرار است که کلاس پشته از آن ارث بری کند.

پراپرتی ها :

Count : از نوع **int** است و نشان دهنده تعداد اعضای موجود در داده ساختار است.

MaxSize : از نوع **int** است و نشان دهنده حداکثر تعداد اعضای موجود در داده ساختار است.

Elements : آرایه ای از نوع داده ای ، متناسب با نوع داده ای اینترفیس ، به اندازه **MaxSize** جهت نگه داری اعضای موجود در داده ساختار است.

متد ها :

Print : این متود به عنوان ورودی ، یک دلگیت با خروجی **bool** و ورودی **int** را دریافت می کند و مواردی که بر اساس دلگیت باید انتخاب شوند را به صورت رشته خروجی می دهد.

کلاس MyStack :

داده ساختار پشته نوع داده ای است که در آن اولین عضو وارد شده، آخرین عضوی است که از آن خارج می‌شود. مثال از آن در دنیای واقعی، تعدادی کتاب است که بر روی هم قرار گرفته اند. به طوری که برای برداشتن کتاب های مراحل پایین تر نیاز است که کتاب های بالاتر را در ابتدا برداریم و سپس به کتاب مورد نظر دست یابیم.

این کلاس جنریک می‌باشد و دو اینترفیس `IData` و `IEnumerable` را پیاده سازی می‌کند.

کانستراکتور : در کانستراکتور این کلاس داده ای از نوع `int` را دریافت می‌کنید که نشان دهنده حداکثر سایز پشته است (`MaxSize`). سپس آرایه ای به این سایز می‌سازید که همان آرایه `Elements` می‌باشد.

پراپرتی ها:

Top : این پراپرتی آخرین عضو پوش شده به استک را به عنوان خروجی بر می‌گرداند.

توجه ۱ : در صورتی که در پشته هیچ عضوی وجود نداشته باشد، خطای مناسب را ایجاد کنید و به کاربر پیام متناسب با آن را نمایش دهید.

متد ها :

Push : این متد با دریافت یک داده، پس از افزایش مقدار `Count` به اندازه یک واحد، آن را به آرایه `Elements` اضافه می‌کند.

توجه ۲ : در صورتی که سایز پشته جهت اضافه کردن داده به اندازه کافی بزرگ نباشد، خطای مناسب را ایجاد کنید و به کاربر پیام متناسب با آن را نمایش دهید.

Pop : این متد آخرین عضو اضافه شده به پشته را از آن حذف می‌کند. بنابراین لازم است که علاوه بر کاهش مقدار `Count` به اندازه یک واحد، آخرین عضو موجود در لیست `Elements` را از آن حذف کنید و به عنوان خروجی بر گردانید.

توجه ۳ : در صورتی که در پشته هیچ عضوی وجود نداشته باشد، خطای مناسب را ایجاد کنید و به کاربر پیام متناسب با آن را نمایش دهید.

Print : این متد را طوری پیاده سازی کنید که اعضای موجود در پشته را از بالا به پایین به صورت رشته برگرداند. به عبارتی نحوه حرکت بر روی اعضای پشته از آخرین عضو وارد شده به طرف اولین عضو آن است.

همانطور که پیش از این اشاره شد این تابع به ورودی یک دلگیت را دریافت می کند که سه روش انتخاب اعضا را مشخص می کند :

۱ . اعضای زوج

۲ . اعضای فرد

۳ . تمام اعضا

نکته ۱ : در پیاده سازی این متد دقت کنید که نمی توانید بر روی اعضای لیست Element پیمایش کنید و بایستی با استفاده از متد های Push و Pop تابع را پیاده سازی کنید.

Clear : پس از اجرای این دستور هیچ داده ای در پشته باقی نخواهد ماند. به عبارتی مقدار Count صفر خواهد شد.

پیاده سازی رابط IEnumerable :

باید به گونه ای پیاده سازی شود که بتوان در یک حلقه بر روی اعضای موجود در پشته پیمایش کرد و مقادیر را به ترتیب خروجشان از استک برگردانیم.

توجه ۴ : این کلاس را به گونه ای پیاده سازی کنید که در هنگام تعریف یک شی از این کلاس بتوان در همان ابتدا داخل پشته مقادیری را اضافه کرد. به عبارتی با استفاده از خط زیر، به ارور کامپایلی بر نخوریم:

```
MyStack<int> s = new MyStack<int>() { 1, 2, 3 };
```

تابع Main برنامه:

برای این بخش در ابتدا از کاربر سائز استک به همراه پنج عضو ابتدایی که قرار است به استک پوش شوند را دریافت کنید. دقت کنید که سائز استک باید از پنج بزرگتر باشد.

سپس منوی زیر را به کاربر نمایش دهید:

1. Push
2. Pop
3. Top
4. Print
5. Exit

با دریافت یک شماره که متناظر با عملیات مورد نظر است، عملیات را انجام داده و خروجی را به کاربر نمایش دهید. (عملیات پوش از آنجا که خروجی ای ندارد، تنها چاپ کردن این که عملیات با موفقیت انجام شده است کافی می باشد.) و تا زمانی که شماره 5 وارد نشده است، برنامه به اجرا شدن ادامه خواهد داد.

تمرین ۴ . شطرنج

با بازی شطرنج و قوانین آشنایی دارید . در این سوال یک بازی شبیه به شطرنج داریم و شما باید در پیاده سازی مفاهیم شی گرای آن خلاقیت داشته باشید .

نکته ۱ . در پیاده سازی این سوال باید از مفاهیم شی گرای و ارث بری استفاده کنید .

توضیحات کلی بازی :

یک زمین شطرنج 8×4 (۸ سطر و ۴ ستون) را در نظر بگیرید . در این بازی فقط ۴ مهره قلعه ، فیل ، اسب و وزیر داریم . در ابتدای بازی این چهار مهره در عرض زمین به ترتیب از راست به چپ : فیل ، اسب ، وزیر و قلعه قرار میگیرند . هر یک از مهره ها در طول بازی ، برای جابجایی در صفحه حرکات منحصر به فردی دارند :

۱ . قلعه : میتواند به صورت افقی و عمودی در صفحه حرکت کند

۲ . فیل : می تواند به صورت قطری در صفحه حرکت کند .

۳ . اسب : میتواند به صورت L (و سایر حالات دورانی L) (2×1) در صفحه حرکت کند .

۴ . وزیر میتواند یکی در میان به شکل قلعه و فیل رفتار کند .

دقت داشته باشید که مهره ها با همان شکلی که در فوق گفته شد فقط میتوانند یک واحد حرکت داشته باشند البته برای اسب همان حرکت 2×1 یک واحد در نظر گرفته میشود .

نحوه بازی :

همانطور که گفته شد در ابتدای بازی این چهار مهره در عرض زمین به ترتیب از راست به چپ : فیل ، اسب ، وزیر و قلعه قرار میگیرند . سپس به صورت رندوم یکی از مهره ها جهت جابجایی انتخاب شده و با توجه به شکل حرکت خود ، در صفحه جابجا میشود . جهت جابجایی آنها در صفحه نیز به صورت رندوم صورت میگیرد . همچنین حرکت آن ها به صورتی است که وضعیت ثانویه مهره در همان ردیف یا ردیف های بالاتر خواهد بود .

توجه ۱ : در صورتی که نوبت مهره (آ) باشد و مهره (ب) در مسیر مهره (آ) باشد ، مهره (آ) میتواند مهره (ب) را زده و خودش در جای (ب) قرار بگیرد .

در پایان مهره ای برنده این بازی خواهد بود که ۸ سطر را زودتر طی کند و به انتهای صفحه برسد .

توجه ۲ : دقت کنید که در ابتدای بازی که همه مهره ها در سطر اول قرار دارند هیچ مهره ای نمی تواند در این سطر مهره دیگری را بزند .

موفق باشد :)