

به نام خدا

با سلام و آرزوی اوقات خوش برای همگی ☺

امتحان پایان ترم برنامه نویسی پیشرفته (6 تیر 1400)

مدت امتحان **120 دقیقه** است و از ساعت 11 تا 13 می باشد.

امتحان شامل 5 سوال می باشد.

باید کدهای خود را در کویرا آپلود کنید.

قسمت هایی که برای چک کد یا کارکرد بهتر نوشته اید را با توضیحات مناسب کامنت کنید.

عدم تسلط به پاسخ های ارایه شده به منزله تقلب خواهد بود.

سوال های خود را در مدت امتحان می توانید در تلگرام یا کلاس درس مطرح کنید.

راه حل هایی که مشابهت غیرمنطقی داشته باشند نمره منفی می گیرند.

به هیچ وجه پاسخ یا ایده خود را در اختیار کسی قرار ندهید.

سوال اول -

یک Extention method به نام countZeros برای نوع داده ای int بنویسید که تعداد رقم های صفر یک مقدار int را برگرداند.

سوال دوم -

کلاس Account شامل فیلد نام از نوع رشته و کد ملی از نوع صحیح و شماره حساب از نوع صحیح و موجودی حساب از نوع اعشاری را بنویسید. سطح دسترسی نباید public باشد ولی دقت کنید ممکن است کلاس های derive (ارث برده از این کلاس) نیاز به دسترسی به این فیلدها داشته باشند.

این کلاس دو تا سازنده دارد. یک سازنده همه موارد را به عنوان پارامتر در ورودی می گیرد. اگر موجودی حساب منفی باشد نباید شی ای ساخته شود و به جاش خطا ایجاد شود. اگر شماره حساب یک عدد چهاررقمی نبود فقط 4 رقم سمت راست آن را در نظر بگیرد (یعنی به جای 12345 عدد 2345 و به جای 23 عدد 0023 در نظر گرفته می شود). اگر شماره حساب در نظر گرفته شده تکراری باشد نباید شی ای ساخته شود و به جاش خطا ایجاد شود

سازنده دیگر کلاس، باید به عنوان پارامتر ورودی نام و کد ملی را بگیرد. همچنین باید یک شماره حساب 4 رقمی رندم تولید کند که از شماره حساب هایی که قبلا ثبت شده اند متفاوت باشد. با استفاده از فراخوانی دیگر سازنده کلاس، شی با موجودی صفر ایجاد کند.

کلاس Babyaccount از کلاس Account به ارث می برد. این کلاس دو فیلد اضافه سقف برداشت از نوع اعشاری و سن از نوع عدد صحیح دارد. همچنین این کلاس باید به گونه ای تعریف شود که کلاس دیگری نتواند از آن به ارث ببرد.

یک سازنده دارد که پارامترهای ورودی اش نام، کد ملی، موجودی حساب، سقف برداشت و سن است. با استفاده از سازنده کلاس base (یعنی به صورت صحیح سازنده تعریف شده در کلاس Account را فراخوانی کند) به فیلدهای کد ملی، موجودی حساب و شماره حساب مقدار می دهد. در بدنه سازنده کلاس، سقف برداشت مقداری می شود به این صورت که صورت منفی بودن شی ایجاد نشود و خطا تولید شود. همچنین شماره حساب باید به این صورت تغییر کند که به ابتدای آن یک عدد یک اضافه شود (شماره حساب های Babyaccount همه یک عدد 5 رقمی هستند که با 1 شروع شده اند). همچنین سن هم باید عددی بین 1 تا 18 باشد در غیر اینصورت خطا ایجاد شود.

سوال سوم -

یک interface به نام Iprintable تعریف کنید که شامل یک متد به نام print باشد که پارامتر ورودی ندارد و خروجی آن String است.

یک تابع جنریک به نام genFun تعریف کنید که یک پارامتر ورودی (به نام t) از نوع جنریکی (مثلاً T1) دارد که نوع آن باید هم Iprintable و هم IEnumerable باشد.

در بدنه تابع genFun باید آرگومان ورودی t را در نظر بگیرد و تعدادش و خودش را در یک فایل با نام OUTXXXX.txt چاپ کند. به جای XXXX باید یک عدد رندم قرار بگیرد. (دقت کنید که t قابلیت اینومریت شدن دارد پس می توانید تعداد اعضایش را بشمارید و همچنین با استفاده از تابع print خودش را چاپ کنید)

سوال چهارم -

یک کالکشن جنریک به نام GenCol بنویسید که بر روی type هایی تعریف شود که interface IComparable را پیاده سازی کرده باشند. فرض کنید پارامتر مورد نظر T1 است.

این کالکشن شامل دو عمل add و remove است.

عمل add دو شی از نوع T1 می گیرد و شی کوچکتر را به مجموعه اضافه می کند.

عمل remove بزرگترین شی مجموعه را حذف می کند و به عنوان خروجی برمیگرداند.

این کالکشن باید IEnumerable باشد به صورتی که اعضای کالکشن را به صورت مرتب شده صعودی enumerate کند.

یک مثال برای درک کالکشن. فرض کنید یک کالکشن GenCol<int> تعریف کنیم و به ترتیب تابع add را با مقادیر 0,8 6,10 3,4 1,1 صدا بزنیم. اگر این کالکشن را در یک Foreach اعضایش را Enumerate کرده و چاپ کنیم در خروجی مقادیر 0 1 3 6 به ترتیب چاپ می شود. همچنین اگر 5 مرتبه تابع remove را صدا بزنیم به ترتیب مقادیر 0 1 3 6 را برگرداند و بار پنجم خطا ایجاد کند.

سوال پنجم -

کلاسی به نام Account داریم که شامل فیلدهای public برای نگهداری نام (از نوع رشته)، کد ملی (از نوع عدد صحیح)، شماره حساب (از نوع عدد صحیح) و موجودی (از نوع اعشاری) است. برای نگهداری حساب های یک بانک، همه حساب های آن در یک لیست به نام infos از نوع Account ذخیره شده اند. با استفاده از linq، پرسش های مناسب (Query)، برای هر کدام از سوالات زیر طراحی کنید.

مجموعه ای شامل کد ملی دارندگان حساب به ترتیب نزولی. (نباید کد ملی تکراری چاپ شود و تغییرات بعدی نباید بر روی خروجی بدست آمده تاثیری بگذارد)

مجموع کل دارایی های همه حساب هایی که دو شماره آخر شماره حساب دو صفر است. (یعنی شماره حساب در سمت راست دو صفر داشته باشد)

دارایی کسی که کمترین مجموع دارایی را دارد. (فرض کنید یک کد ملی میتواند با اسامی مختلف و شماره حساب های مختلف در بانک دارایی داشته باشد.)

دقت کنید پرسش شما بر روی یک مجموعه مثال اعمال می شود و تنها در صورتی نمره میگیرید که خروجی صحیح را برگرداند.