



دانشکده مهندسی کامپیوتر

## برنامه‌سازی پیشرفته

نیم‌سال دوم ۱۴۰۳-۱۴۰۴

## امتحان عملی شماره ۳

سید صالح اعتمادی

ملیحه حاجی حسینی

آرین ثابت

زمان امتحان ۳ ساعت

زمان حضور در سایت ۸:۱۵ صبح

زمان شروع امتحان قبل از ساعت ۸:۳۰ وقتی شرایط آماده شد

چهارشنبه دوازدهم شهریور ماه ۱۴۰۴

## شرایط امتحان

- فقط و فقط یک برگه پشت و رو به عنوان کمک می‌توانید سر امتحان داشته باشید.
- برگه پیش‌نویس خالی به همراه خودکار/مداد می‌توانید همراه‌تون داشته باشید.
- همه وسایل از جمله کاپشن و کیف باید در مرکز سایت قرار بگیرد.
- داشتن خوراکی و/یا نوشیدنی «بی صدا» اشکال ندارد.
- اینترنت شما باید قطع باشد. چنانچه اینترنت کسی وصل باشد، حتی اگر استفاده نکند، تقلب محسوب شده و نمره صفر دریافت کرده و از ادامه امتحان محروم خواهد شد.
- نور لپ‌تاپ شما باید در حالت بیشترین مقدار باشد بصورتیکه مراقب‌های امتحان که از پشت شما عبور می‌کنند به راحتی محتوای صفحه شما را ببینند.
- تنها صفحه‌های مجاز باز روی لپ‌تاپ شما صفحه cmd، فایل پی‌دی‌اف سوال‌ها و وی‌اس‌کد می‌باشد.
- همه اپلیکیشن‌های پیام‌رسان از جمله تلگرام/واتس‌اپ/بله/ایتا/سروش/... باید کامل بسته بوده و حتی در بخش نوتیفیکیشن هم موجود نباشند.
- همه افزون‌های دستیار هوشمند کدنویسی از جمله copilot باید غیر فعال باشند.
- استفاده از هر گونه منبع یا کدی حتی اگر خود شما باشد، مجاز نیست.
- در صورت نیاز به منبع می‌توانید به دو کتاب سی‌شارپی که همراه سوال‌های امتحان ضمیمه شده است مراجعه کنید.

## شروع و پایان

- قبل از شروع امتحان لازم است همه یک پروژه console به نام E3 و یک پروژه تست به نام E3.Test درست کرده هر دو را به یک solution به نام E3.sln اضافه کرده و بیلد و تست کنید. قبل از شروع امتحان و پخش سوال‌ها این مورد برای همه چک می‌شود. چنانچه کسی این شرایط را نداشت، سوال‌ها در اختیار او قرار نمی‌گیرید تا اینکه این شرایط را درست کرده و سپس اینترنت خود را قطع کرده تا بتواند سوال‌ها را دریافت کرده و امتحان را شروع کند.
- همه تست‌ها کامنت شده و `Assert.Inconclusive()` برای آنها قرار داده شده است. **یکی-یکی از حالت کامنت در آورده و پیاده‌سازی‌های لازم برای بیلد شدن را انجام دهید. سپس `Assert.Inconclusive()` را کامنت کرده و است پاس شدن تست اطمینان حاصل کنید.** چنانچه پاس نشد، `Assert.Inconclusive()` را از حالت کامنت در آورده و به سراغ سوال بعدی بروید.
- بعد از پایان زمان امتحان (یا پاس شدن همه تست‌هایی که می‌توانستید تست کنید) بدون سر و صدا دست خود را بالا ببرید تا یکی از مراقب‌های امتحان تست‌های شما را بررسی کرده و نمره‌دهی کند. بعد از آن بدون سر و صدا از سالن خارج شوید.

## نامگذاری‌ها

این امتحان باید:

- در پوشه E3 در ریشه ریپازیتوری گیت پروژه E3-[yourname] (برای عضویت کلیک کنید: <https://classroom.github.com/a/WotrRfeG>)

- نام solution: E3
  - نام پروژه اصلی: E3
  - نام پروژه تست: E3.Tests
  - پروژه باید add/commit/push بشود و بعد از بیلد موفقیت آمیز "کامل" شود.
  - جهت رفع دغدغه دانشجویان تست‌ها در سایت گیت‌هاب، مشابه تمرینات اجرا نخواهد شد (تصحیح و اجرای تست‌ها دستی است) و نحوه قرارگیری پوشه‌ها به ظرافت تمرینات اهمیتی ندارد. بلکه صرفاً قابل اجرا بودن پروژه از طریق سلوشن مورد اهمیت است.
  - برای هر سوال نام تست مورد نظر که باید پاس بشود در متن سوال آمده است.
  - برای پاس‌شدن هر کدام از تست‌ها نام کلاس و متد را با مطالعه تست بیابید. نام فایل‌های کد را به دلخواه خود نام‌گذاری کنید.
- دستورالعمل :** برای هر سوال، کد مورد نیاز را در فضای نام E3 پیاده‌سازی کنید. برای درک کامل نیازمندی‌های هر سوال، به فایل یونیت تست مربوطه در فضای نام E3.Tests مراجعه کنید. موفقیت در گذراندن تست‌ها به منزله پاسخ صحیح است.
- **سوال صفر:** یک کلاس پایه Animal و یک کلاس مشتق شده Dog ایجاد کنید. کلاس Dog باید متد MakeSound را بازنویسی (override) کند. برای جزئیات، به تست Q0\_Inheritance\_ShouldOverrideSound مراجعه کنید.
  - **سوال ۱:** تفاوت بین Value Type (struct) و Reference Type (class) را با پیاده‌سازی کلاس MyPointClass و ساختار MyPointStruct نشان دهید. برای جزئیات، به تست Q1\_ValueVsReferenceType مراجعه کنید.
  - **سوال ۲:** یک کلاس ژنریک به نام Comparer<T> بنویسید که دو آیتم از یک نوع را نگه داشته و بزرگترین آنها را بر اساس پیاده‌سازی IComparable<T> برگرداند. به تست Q2\_GenericsAndIComparable مراجعه کنید.
  - **سوال ۳:** کلاسی به نام ResourceManager طراحی کنید که اینترفیس IDisposable را پیاده‌سازی کند تا نشان دهد یک منبع آزاد شده است. به تست Q3\_IDisposablePattern مراجعه کنید.
  - **سوال ۴:** با استفاده از متدهای Where و Select در LINQ، اعداد زوج بزرگتر از ۵ را فیلتر کرده و سپس دو برابر کنید. به تست Q4\_SimpleLinq مراجعه کنید. در این سوال فقط از توابع LINQ باید استفاده کنید. استفاده از حلقه مجاز نیست.
  - **سوال ۵:** با استفاده از LINQ، لیستی از اشیاء Sale را بر اساس Category گروه‌بندی کرده و مجموع Amount هر گروه را محاسبه کنید. به تست Q5\_Linq\_GroupByAndSum مراجعه کنید. در این سوال فقط از توابع LINQ باید استفاده کنید. استفاده از حلقه مجاز نیست.
  - **سوال ۶:** یک نوع داده Money ایجاد کنید که از اپراتور + برای جمع و اینترفیس IEquatable<T> برای برابری پشتیبانی کند. به تست Q6\_OperatorOverloadingAndIEquatable مراجعه کنید.

- سوال ۷: یک delegate به نام StringOperation تعریف کرده و متدی بنویسید که این نماینده را به عنوان پارامتر برای پردازش رشته‌ها بپذیرد. به تست Q7\_Delegates مراجعه کنید.
- سوال ۸: با استفاده از Func<string, int> و یک عبارت لامبدا (lambda expression)، تابعی بنویسید که طول یک رشته را محاسبه کند. به تست Q8\_LambdaExpressions مراجعه کنید.
- سوال ۹: الگوی Observer را با استفاده از event پیاده‌سازی کنید. کلاس Publisher باید یک رویداد منتشر کند و کلاس Subscriber به آن گوش دهد. به تست Q9\_EventsAndObserverPattern مراجعه کنید.
- سوال ۱۰: حذف شد.
- سوال ۱۱: یک متد بنویسید که یک تابع شمارنده (counter) با استفاده از closure برگرداند. هر بار که تابع بازگشتی فراخوانی می‌شود، باید عدد بعدی را برگرداند. به تست Q11\_Closures مراجعه کنید.
- سوال ۱۲: یک کلاس SafeCounter پیاده‌سازی کنید که چندین ترد (thread) بتوانند به صورت همزمان مقدار آن را افزایش دهند، بدون اینکه تداخل داده (race condition) رخ دهد. از lock استفاده کنید. به تست Q12\_MultithreadingWithLock مراجعه کنید.
- سوال ۱۳: یک متد async بنویسید که یک عملیات طولانی را با Task.Delay شبیه‌سازی کرده و نتیجه‌ای را پس از تاخیر برگرداند. به تست Q13\_AsyncAwait\_ShouldReturnDataAfterDelay مراجعه کنید.
- سوال ۱۴: یک کلاس ژنریک Repository<T> ایجاد کنید که از محدودیت where برای اطمینان از اینکه T دارای یک سازنده پیش‌فرض (new()) و از نوع IEntity است، استفاده کند. به تست Q14\_GenericWhereConstraints مراجعه کنید.
- سوال ۱۵: زنجیره مسئولیت برای تبدیل رشته بسازید (Trim→Upper→Suffix). تست Q15\_ChainOfResponsibility\_Transforms را مطالعه کنید.

موفق باشید