



دانشکده مهندسی کامپیوتر

برنامه‌سازی پیشرفته

نیم‌سال دوم ۱۴۰۳-۱۴۰۴

امتحان عملی شماره ۱

سید صالح اعتمادی

ملیحه حاجی حسینی

آرین ثابت

زمان امتحان ۳ ساعت

زمان حضور در سایت ۷:۳۰ صبح

زمان شروع امتحان قبل از ساعت ۸ وقتی شرایط آماده شد

چهارشنبه سوم اردیبهشت 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 کد خود شما باشد، مجاز نیست.

شروع و پایان

- قبل از شروع امتحان لازم است همه یک پروژه console به نام Exam1 و یک پروژه تست به نام Exam1.Test درست کرده هر دو را به یک solution به نام Exam1.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 نمی‌گیرد 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 El در ریشه ریپازیتوری گیت پروژه exam1-[yourname]

(برای عضویت کلیک کنید: <https://classroom.github.com/a/wjUjycrg>)

- نام solution: E1
- نام پروژه اصلی: E1
- نام پروژه تست: E1.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.

سوال ۱ – جمع دو عدد

کلاس و متد لازم برای پاس تست Q1 را پیاده‌سازی کنید.

سوال ۲ – ضرب و ریست

متد و کلاس لازم برای پاس شدن تست Q2 را پیاده‌سازی کنید. بعد از صدا زدن متد `Q2MultiplyAndReset` مقدار متغیر اول ضرب دو متغیر و متغیر دوم برابر یک باید باشند.

سوال ۳ – بعلاوه یک

نوع `Type1` و `Type2` را به گونه‌ای تعریف کنید که تست Q3 پاس شود. دقت کنید که تابع `Q3Increment` پیاده‌سازی شده است و شما فقط باید این دو نوع را به شکل مناسب تعریف کنید.

سوال ۴ – تعریف فرد

کلاس `Q4Person` را به همراه سازنده (Constructor) و متدها و متغیرهای لازم به شکلی کنید که تست Q4 پاس شود.

سوال ۵ – اکسپشن (Exception)

متد `Q5TryCatchFinally` را به شکلی پیاده‌سازی کنید که تست Q5 پاس شود. راهنمایی: پارامتر اول نشان‌دهنده این است که آیا این متد باید `throw` بکند یا نه. پارامتر دوم کارهای انجام شده در متد را گزارش می‌کند و پارامتر سوم مشخص می‌کند که آیا اکسپشن باید گرفته (catch) شود یا نه. برای گرفتن نمره این سوال لازم است هر دو تست Q5 پاس شوند.

سوال ۶ – سانتیگراد و فارنهایت

کلاس `Q6Temperature` را به گونه‌ای پیاده‌سازی کنید که بدرستی دمای ثبت شده با سانتیگراد یا فارنهایت را به هر دو واحد گزارش دهد.

برای تبدیل دما بین درجه سانتی‌گراد (C°) و فارنهایت (F°) از دو فرمول زیر استفاده می‌شود:
از سانتی‌گراد به فارنهایت

$$32 + \left(\frac{9}{5} \times C^{\circ}\right) = F^{\circ}$$

از فارنهایت به سانتی‌گراد

$$\frac{5}{9} \times (32 - F^{\circ}) = C^{\circ}$$

سوال ۷ – شکل تکراری و اینترفیس (interface)

کلاس‌های Q7Circle، Q7Rectangle و اینترفیس IShape به همراه ShapeUtils.Q7TotalArea را به گونه‌ای پیاده‌سازی کنید که تست Q7 پاس شود.

سوال ۸ – مقایسه گر رشته

کلاس Q8StringLengthComparer باید اینترفیس IComparer<string> را برای مقایسه کردن دو string پیاده‌سازی کند. روش مقایسه بر اساس طول رشته است. طول رشته null کمتر از رشته خالی ("") است. همچنین اگر دو رشته اندازه هم باشند رشته‌ای که با حروف الفبای کوچکتری شروع می‌شود کوچکتر خواهد بود. به مثال‌های تست توجه کنید.

سوال ۹ – عملگر مقایسه برای رشته

کلاس Q8ComparableString را بگونه‌ای پیاده‌سازی کنید که تست Q9 پاس شود. روش مقایسه مانند Q8 است. فقط عملگرهای !=, ==, >, < پیاده‌سازی شده‌اند.

سوال ۱۰ – استراتژی قیمت گذاری

کلاس PricingEngine یک موتور قیمت‌گذاری است که یک استراتژی تخفیف در سازنده (constructor) به عنوان پارامتر دریافت کرده و طبق آن قیمت‌ها را محاسبه می‌کند. قابلیت تغییر استراتژی با متد SetStrategy نیز امکان‌پذیر است. استراتژی‌های «عدم تخفیف» یا «تخفیف درصدی» باید پیاده‌سازی شوند. راهنمایی: لازم است یک اینترفیس مناسب برای استراتژی تخفیف تعریف کرده و بصورت مناسب از آن در پیاده‌سازی PricingEngine و PercentageDiscountStrategy و NoDiscountStrategy استفاده کنید.

موفق باشید