



دانشکده‌ی مهندسی کامپیوتر

برنامه‌سازی پیشرفته (سی شارپ)
آزمون عملی سوم

علی حیدری
استاد: سید صالح اعتمادی

۲۰ خرداد ۱۳۹۸

فهرست مطالب

۳	۱	مقدمه و آماده‌سازی
۳	۱.۱	نکات مورد توجه
۴	۲.۱	آماده‌سازی‌های اولیه
۴	۱.۲.۱	آماده‌سازی‌های مربوط به git
۴	۲.۲.۱	آماده‌سازی‌های مربوط به visual studio
۵	۲	پیاده‌سازی آزمون
۵	۱.۲	مجموعه تست‌های Basics
۵	۱.۱.۲	تست CalculateSum
۵	۲.۱.۲	تست CalculateSumInvalid
۵	۳.۱.۲	تست CalculateSumFormatException
۵	۴.۱.۲	تست TryCalculateSum
۵	۵.۱.۲	تست PIPrecision
۵	۶.۱.۲	تست Fibonacci
۶	۷.۱.۲	تست RemoveDuplicatesDebug
۶	۲.۲	مجموعه تست‌های ThreadsAndEvents
۶	۱.۲.۲	تست GetElapsedTimes
۶	۲.۲.۲	تست DuplicateNumberAdded
۶	۳.۲.۲	تست MakeItFaster
۶	۳.۲	مجموعه تست‌های MessageAnalysis
۶	۱.۳.۲	تست MostRepliedMessage
۶	۲.۳.۲	تست MostPostedMessagePersons
۶	۳.۳.۲	تست MostActiveAtMidNight
۷	۴.۳.۲	تست MostQuestionsWithNoAnswer
۷	۴.۲	مجموعه تست‌های Inheritance
۷	۱.۴.۲	تست Person
۷	۲.۴.۲	تست Student
۷	۳.۴.۲	تست Employee
۷	۴.۴.۲	تست Teacher
۷	۵.۲	مجموعه تست‌های CustomStringOperator
۷	۱.۵.۲	تست StringExplicitOperator
۷	۲.۵.۲	تست StringReverseExplicitOperator
۷	۳.۵.۲	تست StringPlusPlusOperator
۷	۴.۵.۲	تست StringMinusMinusOperator
۸	۳	ارسال تمرین
۸	۱.۳	مشاهده وضعیت اولیه فایل‌ها
۸	۲.۳	اضافه کردن فایل‌های تغییر یافته به stage
۸	۳.۳	commit کردن تغییرات انجام شده
۹	۴.۳	ارسال تغییرات انجام شده به Remote repository
۹	۵.۳	ساخت Pull Request
۹	۶.۳	ارسال Pull Request به بازبیننده

۱ مقدمه و آماده‌سازی

۱.۱ نکات مورد توجه

- برای برگزاری موفقیت‌آمیز و همراه با آرامش امتحان عملی لطفا نکات زیر را رعایت کنید. عدم تذکر من نشانه عدم کشف تخلف نیست.
- دقت کنید که تمامی تست‌ها در ابتدا Comment شده‌اند و شما باید برای اجرا آن‌ها را از این حالت خارج کنید.
- مشکلات یا سوالات خود را فقط از طریق [این فرم](#) اعلام کنید.
- ۱۵ دقیقه قبل از زمان شروع امتحان در محل امتحان حاضر باشید تا از تذکرات و روند امتحان مطلع شوید و هم‌چنین بتوانید برخی از آماده‌سازی‌های اولیه را انجام دهید.
- استفاده از کد تمرین‌های خودتان بلامانع است اما نباید از کد فرد دیگری استفاده کنید.
- ابتدای هر تست `Assert.Inconclusive();` اضافه شده است که باعث می‌شود از اجرای تست شما صرف نظر شود. برای اجرای کامل تست باید این خط را حذف کنید.
- استفاده از ویدیوهای آموزشی حین امتحان مجاز نیست.
- استفاده از تلفن همراه بعد از شروع امتحان مجاز نیست. چنان‌چه قصد استفاده از تلفن همراه برای اتصال به شبکه دارید قبل از شروع امتحان تنظیمات لازم را انجام داده، و پس از قرار دادن در وضعیت بی‌صدا درون کیف خود خارج از دید و دسترس قرار دهید.
- می‌توانید از هر گونه متن کاغذی یا دیجیتالی (کتاب، جزوه، PDF) استفاده کنید.
- پاسخ هر سوال را در مخزن `Git`^۱ خودتان بارگذاری نمایید.
- توجه داشته باشید که برای کسب نمره‌ی قبولی درس کسب حداقل نصف نمره‌ی هر سری تمرین و امتحان الزامی می‌باشد.
- کپی کردن هرگونه کد از روی اینترنت یا غیراینترنت مجاز نیست. پاسخ‌رسانی هر کس حتما باید توسط خود او نوشته شده باشد. کمک گرفتن از دیگران در طول مدت امتحان مجاز نیست و منجر به درج نمره‌ی مردود برای این درس می‌شود.
- حین امتحان تنها اجازه ارتباط با استاد درس را دارید. هر گونه ارتباط با هر فرد دیگری در جلسه امتحان یا خارج از جلسه امتحان به صورت حضوری یا مجازی مجاز نمی‌باشد.
- لطفا تمهیدات لازم برای عدم نیاز به خروج از جلسه امتحان را قبل از امتحان انجام دهید. در صورت نیاز به خروج از محل امتحان قبل از اتمام امتحان، امکان خروج بعد از هماهنگی با استاد به صورت یک نفر، یک نفر هست.
- صدا و صفحه نمایش شما باید از طریق نرم‌افزار `Flashback recorder` به طور کامل از ابتدا تا انتهای امتحان با تنظیمات گفته شده ضبط و ذخیره شود. لطفا قبل از شروع امتحان از درستی عملکرد این نرم‌افزار مطمئن شوید و در صورت وجود اشکال آن را رفع کنید.
- استفاده از خوردنی و نوشیدنی بدون سروصدا و بهم‌زدن نظم جلسه اشکالی ندارد.
- از هرگونه رفتار که باعث برهم‌زدن نظم و حواس‌پرتی دیگران می‌شود (مانند تایپ کردن پرسروصدا با صفحه‌کلید، بازی با قسمت فشاری خودکار، زیاد بودن صدای لپ‌تاپ هنگام خطاهای ویندوز و...) پرهیز کنید.
- استفاده از اینترنت فقط برای استفاده از `Git` و `visualstudio.com` لازم است. خارج از این تنها استفاده جایز از اینترنت برای پیدا کردن `syntax` سی‌شارپ یا کمک در رابطه با خطای کامپایل از سایت `MSDN` می‌باشد. از این جهت کلیه‌ی اتصالات مشکوک بررسی و پی‌گیری می‌شوند.
- با توجه به این‌که به هیچ سوالی پاسخ داده نمی‌شود مسئولیت رفع تمامی مشکلات پیش‌آمده با خود شماست. بنابراین توصیه می‌شود که پیش از شروع امتحان دستورات زیر را در مخزن `Git` محلی^۲ خود وارد کنید تا بتوانید اشکالات احتمالی را پیش از شروع امتحان رفع کنید.
- ابتدا با دستور `git status` از وضعیت کنونی مخزن مطلع شوید. در صورت وجود فایل یا پوشه‌های `commit` نشده آن‌ها را در شاخه^۳ مناسب `commit` کنید.
- مثلا در حالت زیر پوشه‌ی `A13` در وضعیت `stage` نشده قرار دارد و شما باید ابتدا آن را به `stage` اضافه و سپس آن را `commit` کنید. دقت کنید که پوشه و فایل‌های مربوط به هر تمرین در شاخه مربوطه‌اش `commit` شود.

^۱Repository

^۲local

^۳branch

```

1 Ali@DESKTOP-GS7PR56 MINGW64 /c/git/AP97982 (fb_A13)
2 $ git status
3 On branch fb_A13
4 Untracked files:
5   (use "git add <file>..." to include in what will be committed)
6
7   A13/
8
9 nothing added to commit but untracked files present (use "git add" to track)

```

پس از رفع تمام اشکالات احتمالی به شاخه‌ی `master` رفته و دستور `git pull origin master` را بزنید. این دستور شاخه `master` مخزن remote را روی شاخه `master` مخزن محلی ادغام^۴ می‌کند.

سپس یک‌بار دیگر دستور `git status` را وارد کنید تا از وضعیت مخزن مطلع شوید. خروجی این دستور باید دقیقاً به صورت زیر باشد در صورتی که این خروجی را دریافت نمی‌کنید مخزن شما هنوز آماده نیست و ممکن است در طول امتحان به مشکل بر بخورید.

```

1 Ali@DESKTOP-GS7PR56 MINGW64 /c/git/AP97982 (master)
2 $ git status
3 On branch master
4 nothing to commit, working tree clean

```

۲.۱ آماده‌سازی‌های اولیه

قواعد نام‌گذاری آزمون را از جدول ۱ مطالعه کنید.

جدول ۱: قراردادهای نام‌گذاری آزمون

Naming conventions					
Branch	Directory	Solution	Project	Test Project	Pull Request
fb_E2	E2	E2	E2	E2Tests	Exam2

۱.۲.۱ آماده‌سازی‌های مربوط به git

✓ ابتدا بار دیگر از وضعیت مخزن مطلع شوید. طبق توضیحات گفته شده برای آماده‌سازی‌های مربوط به پیش از امتحان خروجی باید به صورت زیر باشد.

```

1 Ali@DESKTOP-GS7PR56 MINGW64 /c/git/AP97982 (master)
2 $ git status
3 On branch master
4 nothing to commit, working tree clean

```

✓ یک شاخه‌ی جدید با نام `fb_E2` بسازید و تغییر شاخه دهید.

```

1 Ali@DESKTOP-GS7PR56 MINGW64 /c/git/AP97982 (master)
2 $ git checkout -b fb_E2
3 Switched to a new branch 'fb_E2'
4 Ali@DESKTOP-GS7PR56 MINGW64 /c/git/AP97982 (fb_E2)
5 $

```

توصیه می‌شود پس از پیاده‌سازی هر کلاس تغییرات انجام شده را `commit` و `push` کنید.

۲.۲.۱ آماده‌سازی‌های مربوط به visual studio

ساختار فایل پایه‌ای که در اختیار شما قرار می‌گیرد به صورت زیر است:

```

1 E2
2 +---Project

```

⁴merge

```

3 | | Basics.cs
4 | | DotNetInterfaces.cs
5 | | Events.cs
6 | | Inheritance.cs
7 | | Threading.cs
8 | |
9 | \---Linq
10 |     MessageAnalysis.cs
11 |     MessageData.cs
12 |
13 | \---ProjectTests
14 |     BasicsTests.cs
15 |     chats.csv
16 |     CustomStringOperatorTests.cs
17 |     InheritanceTests.cs
18 |     MessageAnalysisTests.cs
19 |     ThreadsAndEvents.cs

```

در فایل پایه دو پوشه وجود دارد شما باید فایل(های) موجود در پوشه Project را به پروژه‌ی اصلی (E2) و فایل(های) موجود در پوشه ProjectTests را به پروژه‌ی تست (E2Tests) اضافه کنید.

۲ پیاده‌سازی آزمون

۱.۲ مجموعه تست‌های Basics

هدف این بخش سنجش آشنایی ابتدایی با پردازش رشته‌ها، پارامترهای خروجی و مدیریت خطا می‌باشد.

۱.۱.۲ تست CalculateSum

متد CalculateSum را بگونه‌ای پیاده‌سازی کنید که یک رشته شامل تعداد عدد و عملگر جمع را به عنوان پارامتر دریافت کرده و مقدار معادل عددی را برگرداند. $\frac{21}{1}$

۲.۱.۲ تست CalculateSumInvalid

در صورتیکه رشته ورودی یک عبارت ریاضی صحیح نباشد لازم است متد CalculateTest یک Exception از نوع InvalidDataException پرتاب کند. $\frac{20}{2}$

۳.۱.۲ تست CalculateSumFormatException

همچنین لازم است در صورتیکه زیررشته موجود در عبارت بین علامت‌های جمع از نوع عدد صحیح نباشد Exception متناظر آن رخ دهد. $\frac{19}{3}$

۴.۱.۲ تست TryCalculateSum

متد TryCalculate را به گونه‌ای پیاده‌سازی کنید که در صورت بروز خطا false برگرداند و در هیچ شرایطی Exception رخ ندهد. در این متد مقدار عددی محاسبه شده بصورت یک پارامتر از نوع out بازگردانده می‌شود. $\frac{18}{4}$

۵.۱.۲ تست PIPrecision

یکی از راه‌های محاسبه عدد پی استفاده از دنباله زیر است.

$$1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots = \frac{\pi}{4}.$$

سوال این است که برای محاسبه عدد پی تا هفت رقم معنادار این دنباله را تا عبارت چندم باید محاسبه کنیم. راهنمایی: Math.PI عدد تا دقت بیش از ده رقم اعشار. :Math.Round(x, 10) عدد ورودی را به ده رقم معنادار رند می‌کند. حال متد PIPrecision را به گونه‌ای پیاده‌سازی کنید که پاسخ این سوال را پیدا کند. $\frac{17}{5}$

۶.۱.۲ تست Fibonacci

Extension متد به نام Fibonacci را برای نوع داده‌ای int به گونه‌ای پیاده‌سازی کنید که عدد n در دنباله بیبوناچی را برگرداند و این تست پاس شود. ^{۱۶/۶}

۷.۱.۲ تست RemoveDuplicatesDebug

کلاس FullName متد RemoveDuplicates پیاده‌سازی شده‌اند ولی این تست پاس نمی‌شود. کلاس FullName را بگونه‌ای تغییر دهید که تست زیر پاس شود. ^{۱۵/۷}

۲.۲ مجموعه تست‌های ThreadsAndEvents

تست‌های این بخش مربوط به IEnumerable/foreach/yield و Event و Thread هستند

۱.۲.۲ تست GetElapsedTimes

لازم است متد GetElapsedTimes به گونه‌ای پیاده‌سازی شود که فاصله دسترسی به مقداری این مجموعه را برگرداند. به این شکل که مقدار اولی که بر میگرداند همیشه صفر است و مقادیر بعدی برابر با تعداد میلی ثانیه‌ای است که از زمان دسترسی به مقدار قبلی گذشته است. پارامتر ورودی این متد حداکثر طول این رشته است. در این متد می‌توانید از کلاس Stopwatch برای اندازه‌گیری زمان استفاده کنید. برای شیوه استفاده از آن به MSDN یا تست‌های پیاده‌سازی شده از که این کلاس استفاده می‌کنند مراجعه کنید. ^{۱۴/۸}

۲.۲.۲ تست DuplicateNumberAdded

کلاس DuplicateNumberDetector را به گونه‌ای پیاده‌سازی کنید که یک متد AddNumber داشته باشد که یک عدد صحیح به عنوان ورودی دریافت می‌کند. چنانچه این متد با یک عدد تکراری صدا زده شود لازم است Event به نام DuplicateNumberAdded اتفاق بیافتد که به عنوان پارامتر تعداد کل دفعاتی که عدد تکراری اضافه شده را برگرداند. ^{۱۳/۹}

۳.۲.۲ تست MakeItFaster

متد MakeItFasterTest را به گونه‌ای پیاده‌سازی کنید که تعدادی Action از ورودی دریافت کرده و آنها را همزمان اجرا کند. این متد تنها زمانی باید برگردد که تمام delegate ها اجرا شده باشند. ^{۱۲/۱۰}

۳.۲ مجموعه تست‌های MessageAnalysis

فایل chats.csv حاوی پیام‌های گروه تلگرامی درس AP این ترم به شما داده شده است. مشابه تمرین ۱۲ VisualBasic Reference را به پروژه‌ی اصلی (E2) اضافه کنید. کلاس‌های لازم برای پارس کردن مشابه با تمرین ۱۲ به شما داده شده است.

۱.۳.۲ تست MostRepliedMessage

برای پاس شدن این تست شما باید از بین کل پیام‌های داده شده پیامی را که بیشترین پاسخ^۵ را دریافت کرده در قابل یک شی از نوع MessageData برگردانید. ^{۱۱/۱۱}

۲.۳.۲ تست MostPostedMessagePersons

برای پاس شدن این تست شما باید ۵ نفری را که بیشترین تعداد پیام را در گروه ارسال کرده‌اند به جز استاد "Saeleh Eetemadi" و سرتی‌ای "Ali Heydari" به صورت یک آرایه از زوج مرتبها^۶ برگردانید. عضو اول این زوج مرتب نام نویسنده‌ی پیام، و عضو دوم آن زوج مرتب تعداد کل پیام‌هایی است که آن فرد در گروه ارسال کرده می‌باشد. ^{۱۰/۱۲}

^۵reply

^۶Tuple

۳.۳.۲ تست MostActiveAtMidNight

برای پاس شدن این تست شما باید ۵ نفر اولی که بیشترین تعداد پیام را در ساعات اولیه‌ی بامداد 00:00 تا 04:00 ارسال کرده‌اند به صورت آرایه‌ای از زوج مرتب‌ها برگردانید. عضو اول این زوج مرتب نام نویسنده‌ی پیام، و عضو دوم آن زوج مرتب تعداد کل پیام‌هایی است که آن فرد در گروه ارسال کرده می‌باشد. ۹/۱۳

۴.۳.۲ تست MostQuestionsWithNoAnswer

با فرض این که هر پیامی که دارای کاراکتر علامت سوال (هم کاراکتر فارسی علامت سوال '؟' و هم کاراکتر انگلیسی علامت سوال '?') باشد یک سوال محسوب می‌شود کدام فرد بیشترین تعداد سوال بدون پاسخ را پرسیده است؟ نام آن فرد را برگردانید. ۸/۱۴

۴.۲ مجموعه تست‌های Inheritance

کلاس‌های Person، Employee، Student، Teacher باید به گونه‌ای پیاده‌سازی شوند که ارث‌بری‌ها، متدهای virtual، abstract، override به درستی و مطابق با تست‌ها تعریف شوند.

۱.۴.۲ تست Person

Person یک کلاس abstract است با یک سازنده که نام و مونث بودن را به عنوان پارمتر دریافت می‌کند. این کلاس یک virtual getter به نام Name دارد که نام با پیشوند خانم یا آقا را برمی‌گرداند. علاوه بر این یک getter به نام IsFemale نیز دارد. همچنین دارای یک abstract getter به نام LunchRate ۷/۱۵

۲.۴.۲ تست Student

کلاس Student از Person به ارث برده و نرخ ناهار را برابر ۲۰۰۰ تومان قرار می‌دهد. این کلاس را بگونه‌ای پیاده‌سازی کنید که تست زیر پاس شود. ۶/۱۶

۳.۴.۲ تست Employee

کلاس Employee از Person به ارث برده و نرخ ناهار را برابر ۵۰۰۰ تومان قرار می‌دهد. علاوه بر این یک متد virtual به نام CalculateSalary اضافه می‌کند که حقوق کارمند را برابر ساعتی ۵۰۰۰ تومان حساب می‌کند. این کلاس را بگونه‌ای پیاده‌سازی کنید که تست زیر پاس شود. ۵/۱۷

۴.۴.۲ تست Teacher

کلاس Teacher از Employee به ارث برده و نرخ ناهار را برابر ۱۰۰۰۰ تومان قرار می‌دهد. علاوه بر این متد CalculateSalary را override می‌کند که حقوق استاد را برابر ساعتی ۲۰۰۰۰ تومان حساب می‌کند. این کلاس را بگونه‌ای پیاده‌سازی کنید که تست زیر پاس شود. ۲/۱۸

۵.۲ مجموعه تست‌های CustomStringOperator

کلاس MyString باید بگونه‌ای پیاده‌سازی شده و اپراتورهای لازم برای آن overload شده یا متدهای لازم override شوند که تست‌های این مجموعه پاس بشوند.

۱.۵.۲ تست StringExplicitOperator

اپراتور تبدیل نوع داده‌ای از System.String به MyString را پیاده‌سازی کرده و متدها و اپراتورهای لازم برای مقایسه این دو نوع را به گونه‌ای پیاده سازی کنید که این تست پاس شود. ۳/۱۹

۲.۵.۲ تست StringReverseExplicitOperator

اپراتور تبدیل نوع داده‌ای از MyString به System.String را پیاده‌سازی کرده و متدها و اپراتورهای لازم برای مقایسه این دو نوع را به گونه‌ای پیاده سازی کنید که این تست پاس شود. ۲/۲۰

۳.۵.۲ تست StringPlusPlusOperator

اپراتور ++ را برای کلاس MyString به گونه‌ای پیاده‌سازی کنید که رشته حرفی را به حروف بزرگ تبدیل کند. ۱/۲۱

۴.۵.۲ تست StringMinusMinusOperator

اپراتور ++ را برای کلاس MyString به گونه‌ای پیاده‌سازی کنید که رشته حرفی را به حروف کوچک تبدیل کند. ۰/۲۲

۳ ارسال تمرین

در اینجا یک‌بار دیگر ارسال تمرینات را با هم مرور می‌کنیم:

۱.۳ مشاهده‌ی وضعیت اولیه‌ی فایل‌ها

ابتدا وضعیت فعلی فایل‌ها را مشاهده کنید:

```
1 Ali@DESKTOP-GS7PR56 MINGW64 /c/git/AP97982 (fb_E2)
2 $ git status
3 On branch fb_E2
4 Untracked files:
5   (use "git add <file>..." to include in what will be committed)
6
7   E2/
8
9 nothing added to commit but untracked files present (use "git add" to track)
```

همان‌طور که مشاهده می‌کنید فولدر E2 و تمام فایل‌ها و فولدرهای درون آن در وضعیت Untracked قرار دارند و همان‌طور که در خط آخر خروجی توضیح داده شده برای commit کردن آن‌ها ابتدا باید آن‌ها را با دستور git add وارد stage کنیم.

۲.۳ اضافه کردن فایل‌های تغییر یافته به stage

حال باید فایل‌ها و فولدرهایی را که در stage قرار ندارند را وارد stage کنیم. برای این کار از دستور git add استفاده می‌کنیم.

```
1 Ali@DESKTOP-GS7PR56 MINGW64 /c/git/AP97982 (fb_E2)
2 $ git add E2/*
```

حال دوباره وضعیت فایل‌ها و فولدرها را مشاهده می‌کنیم:

```
1 Ali@DESKTOP-GS7PR56 MINGW64 /c/git/AP97982 (fb_E2)
2 $ git status
3 On branch fb_E2
4 Changes to be committed:
5   (use "git reset HEAD <file>..." to unstage)
6
7   new file:   E2/E2.sln
8   new file:   E2/E2/E2.csproj
9   new file:   E2/E2/App.config
10  new file:   E2/E2/Program.cs
11  new file:   E2/E2/Properties/AssemblyInfo.cs
12  new file:   E2/E2Tests/E2Tests.csproj
13  new file:   E2/E2Tests/Properties/AssemblyInfo.cs
14  new file:   E2/E2Tests/packages.config
15  .
16  .
17  .
```

همان‌طور که مشاهده می‌کنید فولدر E2 و تمام فولدرها و فایل‌های درون آن (به جز فایل‌هایی که در gitignore معین کرده‌ایم) وارد stage شده‌اند.

۳.۳ commit کردن تغییرات انجام شده

در گام بعدی باید تغییرات انجام شده را `commit` کنیم. فراموش نکنید که فقط فایل‌هایی را می‌توان `commit` کرد که در `stage` قرار داشته باشند. با انتخاب یک پیام مناسب تغییرات صورت گرفته را `commit` می‌کنیم:

```
1 Ali@DESKTOP-GS7PR56 MINGW64 /c/git/AP97982 (fb_E2)
2 $ git commit -m "Implement Exam2"
3 [fb_E2 c1f21df] Implement Exam2
4 15 files changed, 595 insertions(+)
5 create mode 100644 E2/E2.sln
6 create mode 100644 E2/E2/E2.csproj
7 create mode 100644 E2/E2/App.config
8 create mode 100644 E2/E2/Program.cs
9 create mode 100644 E2/E2/Properties/AssemblyInfo.cs
10 create mode 100644 E2/E2Tests/E2Tests.csproj
11 create mode 100644 E2/E2Tests/Properties/AssemblyInfo.cs
12 create mode 100644 E2/E2Tests/packages.config
13 .
14 .
15 .
```

۴.۳ ارسال تغییرات انجام شده به Remote repository

گام بعدی ارسال تغییرات انجام شده به مخزن Remote است.

```
1 Ali@DESKTOP-GS7PR56 MINGW64 /c/git/AP97982 (fb_E2)
2 $ git push origin fb_E2
3 Enumerating objects: 25, done.
4 Counting objects: 100% (25/25), done.
5 Delta compression using up to 8 threads
6 Compressing objects: 100% (22/22), done.
7 Writing objects: 100% (25/25), 9.56 KiB | 890.00 KiB/s, done.
8 Total 25 (delta 4), reused 0 (delta 0)
9 remote: Analyzing objects... (25/25) (5 ms)
10 remote: Storing packfile... done (197 ms)
11 remote: Storing index... done (84 ms)
12 To https://9752XXX.visualstudio.com/AP97982/_git/AP97982
13 * [new branch] fb_E2 -> fb_E2
```

۵.۳ ساخت Pull Request

با مراجعه به سایت `Azure DevOps` یک Pull Request جدید با نام `Exam2` بسازید به طوری که امکان `merge` کردن شاخه‌ی `fb_E2` را بر روی شاخه‌ی `master` را بررسی کند. (این کار در صورتی انجام می‌شود که کد شما کامپایل شود و همچنین تست‌های آن پاس شوند) در نهایت با انتخاب گزینه‌ی `set auto complete` در صفحه‌ی Pull Request مربوطه تعیین کنید که در صورت وجود شرایط `merge` این کار انجام شود. دقت کنید که گزینه‌ی `Delete source branch` نباید انتخاب شود.

۶.۳ ارسال Pull Request به بازبیننده

در نهایت Pull Request ساخته شده را برای بازبینی، با بازبیننده‌ی خود به اشتراک بگذارید.