

به نام خدا

# آزمون پایان ترم گروه ۱- پاسخنامه

مهندسی نرم افزار - نیمسال اول ۴۰۲

➤ تاریخ آزمون: دوشنبه ۲۵ / ۱۰ / ۴۰۲

➤ زمان شروع: ۱۵:۰۰

➤ مدت زمان آزمون: ۱۵۰ دقیقه



دانشکده مهندسی کامپیوتر - دانشگاه شریف

مدرس: دکتر مهران ریواده

۱- یک شرکت نرم افزاری به تازگی محصول جدید خود را منتشر کرده است. بازخوردهایی گزارش شده که مشکلاتی از جمله دو مورد زیر را بیان می کند. مدیریت احتمال می دهد که عدم توجه به عوامل کیفی خاص علت اصلی می باشد. برای هر کدام از این موارد، ابتدا مشخص نمایید که کدام یک از عوامل کیفیت نرم افزار مک کال (McCall) دیده نشده است؟ سپس دو سیاست یا راهبرد پیشنهادی برای رفع این مشکلات را بیان نمایید.

A. شکایات زیادی از سمت کاربران گزارش شده که نرم افزار در هنگام استفاده خیلی Crash می کند و داده ها از بین می روند.  
B. تعدادی از نیروهای فنی می گویند معماری محصول پیچیده است و انجام آزمون برای آن، چالش برانگیز است.

پاسخ:

#### بخش A سوال:

در مشکل مطرح شده ی A، عامل Reliability وجود ندارد و کاربران در دسترسی به برنامه دچار مشکل شده اند. برای حل این مشکل، می توان کارهای زیر را انجام داد:

- ۱) برنامه تست جامع (Comprehensive testing program): اجرای تست های منظم و گسترده قبل و بعد از انتشار برای شناسایی و رفع خطاهای ممکن.
- ۲) پایش مستمر (Continuous monitoring): استفاده از ابزارهای Monitoring برای شناسایی زود هنگام مشکلات و اقدامات پیشگیرانه.
- ۳) سیستم های گزارش دهی قدرتمند (Robust reporting system): ایجاد فضایی برای ثبت گزارشات دقیق از خطاها توسط کاربران برای تجزیه و تحلیل دقیق و رفع سریع.
- ۴) به روزرسانی های منظم (Regular updates): انتشار نسخه های جدید و اصلاحات برای رفع خطاهای شناخته شده و بهبود عملکرد.
- ۵) ارتباط شفاف با مشتریان (Transparent communication with customers): اطلاع رسانی به مشتریان درباره مشکلات شناسایی شده و اقدامات انجام شده برای رفع آن ها.

#### بخش B سوال:

در ارتباط با بخش B، عامل Testability وجود ندارد. راهکارهای ممکن برای حل این مسئله شامل موارد زیر هستند:

- ۱) ساده سازی و بهبود معماری: در صورت امکان، تلاش برای ساده سازی معماری نرم افزار برای تسهیل تست می تواند مفید باشد.

- ۲) استفاده از تکنیک های تست پیشرفته: به کارگیری تکنیک های تست مانند تست واحد، تست ادغام، و تست غیر عملکردی برای پوشش بخش های مختلف نرم افزار.
- ۳) استخدام Tester های متخصص: به کارگیری Tester های با تجربه و مهارت کافی برای شناسایی و رفع خطاهای پیچیده.

- ۴) استفاده از ابزارهای خودکارسازی تست (Test automation): استفاده از ابزارهای خودکارسازی تست برای کاهش زمان و هزینه تست و افزایش تداوم آن.

- ۵) توجه به تست در مراحل اولیه توسعه: در نظر گرفتن تست به عنوان بخشی جدایی ناپذیر از مراحل توسعه نرم افزار و ادغام تست در چرخه توسعه.

(صفحه ۴۱۶ و ۴۱۷ - بخش ۲-۲-۱۹ - کتاب پرسمن ادیشن ۸)

۲- فرض کنید یک تیم، پروژه‌ای به نام «خانه‌ی امن» را در دست دارد. مستندات مدل نیازمندی‌های این پروژه شامل ۳۰ صفحه است. بازبینی فنی (Technical Review) ۱۶ خطای کوچک (Minor Error) و ۴ خطای بزرگ (Major Error) را در این مدل مشخص کرده است.

A. اگر میزان تراکم خطای میانگین (Error density) این تیم برای مدل نیازمندی‌ها برابر با تراکم خطای مدل نیازمندی‌های

پروژه‌ی «خانه‌ی امن» باشد، این مقدار را به دست آورید و توضیح دهید که مقدار به دست آمده به چه معنا است؟

B. تیم برای تصحیح یک خطای کوچک مدل نیازمندی، بلافاصله پس از بازبینی به ۳ نفر-ساعت نیاز دارد. همچنین برای

یک خطای بزرگ مدل نیازمندی به ۱۵ نفر-ساعت نیاز است. با بررسی داده‌های بازبینی تیم، می‌دانیم خطاهای کوچک

حدود ۵ برابر بیش‌تر از خطاهای بزرگ رخ می‌دهند. خطاهای مربوط به نیازمندی‌ها که در طول آزمون کشف می‌شوند، به

طور متوسط به ۳۷ نفر-ساعت برای یافتن و تصحیح نیاز دارند. تفاوت تلاشی که تیم برای خطاهای مربوط به نیازمندی‌ها

می‌گذارد بین دو حالت داشتن بازبینی فنی و نداشتن آن در پروژه‌ی «خانه‌ی امن» چگونه خواهد بود؟

C. با توجه به مثال بخش قبل و دریافته‌های خود از بازبینی فنی، کدام نمودار برای پروژه‌ای با بازبینی و کدام برای بدون

بازبینی است؟ چرا؟

پاسخ:

بخش A سوال:

با توجه به اطلاعات داده شده در سوال داریم:

$$wps = 30 \text{ pages}$$

$$Err_{minor} = 16 \Rightarrow Err_{tot} = 20$$

$$Err_{major} = 4$$

$$Error \text{ density} = Err_{tot} WPS = \frac{20}{30} = 0.6$$

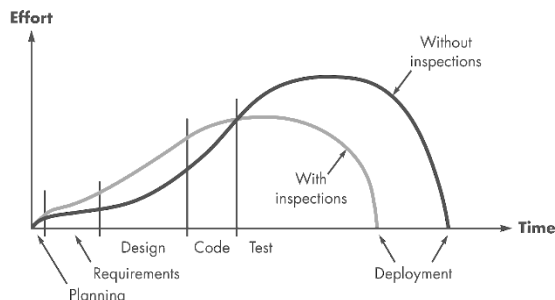
مقدار به دست آمده به این معنی است که به ازای هر صفحه از مستندات مدل نیازمندی ۰,۶ خطا خواهیم داشت. یا به عبارتی در هر ۳ صفحه، ۲ خطا خواهد بود. همچنین با توجه به این که این مقدار برابر با تراکم خطای میانگین این تیم برای مدل نیازمندی‌ها است، تیم می‌تواند با توجه به این مقدار هر گاه بازبینی فنی انجام می‌دهد، بررسی کند که تعداد خطاهایی که یافته، منطقی است یا خیر. اگر تعداد خطاها کم‌تر از میانگین باشد، آنگاه یا برخی از خطاها پیدا نشده‌اند، یا تیم مدل نیازمندی‌ها را بسیار خوب و بهتر از حالت معمولش آماده کرده است.

بخش B سوال:

با توجه به اطلاعات داده شده در سوال، برای زمان میانگین مورد نیاز برای تصحیح خطاهای مدل نیازمندی که در بازبینی پیدا شده باشند به نفر-ساعت داریم:

$$\frac{(5 \times 3) + 15}{5 + 1} = 6$$

از آن جا که برای خطاهای مربوط به نیازمندی‌ها که در طول آزمون کشف می‌شوند، به طور متوسط به ۳۷ نفر-ساعت برای یافتن و تصحیح نیاز است، پس با بازبینی، ۳۲ نفر-ساعت تلاش کم‌تری برای هر خطا نیاز خواهد بود. از آن جا که  $Error_{tot} = 20$  بنابراین اگر تیم بازبینی فنی انجام دهد، در مجموع ۶۴۰ نفر-ساعت تلاش کم‌تری برای خطاهای مدل نیازمندی‌ها نیاز خواهد داشت.

**بخش C سوال:**

همان طور که در مثال قبل هم دیدیم، وقتی از بازبینی استفاده می‌کنیم، ممکن است تا قبل از شروع آزمون (Testing)، تلاش بیش‌تری نیاز باشد، ولی بعد از آن نه تنها تلاش مورد نیاز بسیار کم‌تر خواهد شد، که زمان کوتاه‌تری برای آزمون صرف می‌شود و در نتیجه Deployment زودتر اتفاق می‌افتد.

(صفحه ۴۳۵ تا ۴۳۷ - بخش ۳-۲۰ - کتاب پرسمن ادیشن ۸)

۳- یک تیم ایجاد نرم‌افزار که بر اساس متدولوژی Scrum کار می‌کند با مسائلی رو به رو شده است. با توجه به توصیفی که برای هر یک از مسائل در ادامه آورده شده، به سوال مطرح شده پاسخ دهید.

A. خواسته ذینفعان در فرآیند استخراج نیامندی‌ها این بوده که ورود (Login) به نرم‌افزار بر اساس تکنولوژی رمز یکبار مصرف باشد؛ یعنی کاربر پس از وارد کردن شماره موبایل خود با رمز یکبار مصرفی که از طریق پیامک برای وی ارسال می‌شود، وارد شود. پس از بالا آمدن محصول و ارائه‌ی آن به ذینفعان، مشاهده می‌شود که فرآیند ورود با رمز عبور ایستا انجام می‌شود نه رمز یکبار مصرف؛ به عبارت دیگر کاربر باید یک رمز ۸ کاراکتری برای ورود خود تنظیم کند؛ با اینکه تیم ایجاد این Feature را پیاده‌سازی کرده، آزمون‌کننده (Tester) تیم هم Unit Test را برای آن نوشته، تست کرده و تایید کرده است که فرآیند ورود به درستی کار می‌کند. اما ذینفعان متعجب شده و رضایت ندارند. به نظر شما این اتفاق بیانگر نتیجه کدام یک از مفاهیم حوزه‌ی آزمون است؟ چرا؟ چه کار اشتباه دیگری نیز در این تیم وجود دارد؟

B. ذینفعان تصمیم گرفته‌اند تا بدانند این نرم‌افزار تا چند Login همزمان کاربر را پشتیبانی خواهد کرد و یک پاسخ عددی از تیم می‌خواهند. چه نوع تستی می‌تواند چنین پاسخی را برای آن‌ها فراهم نماید؟ آن تست را توضیح دهید.

**پاسخ:**

**بخش A سوال:**

اولین مفهومی که در این مساله مشاهده می‌شود Valid نبودن Feature پیاده‌سازی شده است. می‌دانیم که Validation به این معناست که نرم‌افزار مطابق با نیازمندی‌های مشتری باشد. در این سناریو، تیم ایجاد، Feature را بر اساس انتظار ذینفعان پیاده‌سازی نکرده است. گرچه فرآیند ورود با رمز ۸ کاراکتری تست گردیده و به درستی رفتار می‌کند، اما کل این Feature از آنجا که با خواسته ذینفعان یکسان نیست، Valid نمی‌باشد. بنابراین تست اعتبارسنجی ویژگی، مشخص نموده که ویژگی پیاده‌سازی شده نادرست است.

مشکل دیگری که در این تیم وجود دارد، اجرای تست واحد (Unit test) توسط آزمون‌کننده تیم است. تست واحد توسط برنامه‌نویس (Developer) نوشته و اجرا می‌گردد. آزمون‌کننده باید کل ویژگی را بر اساس نیازمندی تعریف شده برای آن تست کند.

**بخش B سوال:**

برای پاسخ به این خواسته ذینفعان از Stress Testing می‌توان استفاده نمود. در این نوع تست، نرم‌افزار را در یک شرایط غیرطبیعی و استفاده افراطی قرار می‌دهند تا تعداد درخواست‌ها از منابع آن بالا برود، تا نقاط ضعف و آسیب‌پذیری آن را در شرایط استرسی بالا اندازه‌گیری شده و مشخص کنند که سطح تحمل استرس نرم‌افزار چه اندازه است.

(اسلایدهای آزمون - فصل ۲۲ و ۲۳ و ۲۴ - اسلایدهای ۷ و ۸ و ۱۹)

۴- برنامه زیر را در نظر بگیرید، این برنامه یک عدد صحیح را به عنوان ورودی گرفته و مشخص می کند که عدد داده شده اول است یا خیر.

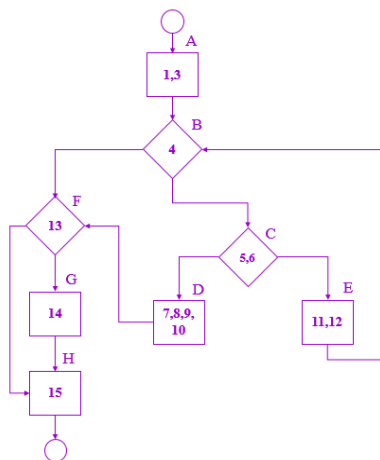
A. گراف کنترل جریان برنامه داده شده را رسم نمایید.

B. با استفاده از تکنیک آزمون مسیرهای پایه (Basis Path Testing) ابتدا مسیرهای اصلی را بدست آورده و سپس برای این برنامه موارد آزمون (Test Case) را طراحی نمایید.

```
int main()
{
    int n, index;
1   cout << "Enter a number: " << n;
3   index = 2;
4   while (index <= n - 1)
5   {
6       if (n % index == 0)
7       {
8           cout << "It is not a prime number" << endl;
9           break;
10      }
11      index++;
12  }
13  if (index == n)
14      cout << "It is a prime number" << endl;
15 } // end main
```

پاسخ:

ابتدا گراف یا فلوچارت قطعه کد برنامه را می کشیم. فلوچارت آن مطابق با شکل مقابل است:



سپس پیچیدگی Cyclomatic آن را محاسبه می کنیم:

$$V(G) = \text{Number of simple descisions} + 1 \Rightarrow 3 + 1 = 4$$

با توجه به این که  $V(G) = 4$  است، بنابراین ۴ مسیر مستقل داریم. که عبارتند از:

- 1) A - B - F - G - H
- 2) A - B - F - H
- 3) A - B - C - E - B - F - G - H
- 4) A - B - C - D - F - H

حال برای هر یک از مسیرهای بدست آمده یک مورد آزمون (Test Case) طراحی می کنیم.

| شماره مورد آزمون | مقدار ورودی (n) | مقدار خروجی | مسیر پوشش داده شده            |
|------------------|-----------------|-------------|-------------------------------|
| ۱                | ۱               | ندارد       | A - B - F - H                 |
| ۲                | ۲               | اول         | A - B - F - G - H             |
| ۳                | ۳               | اول         | A - B - C - E - B - F - G - H |
| ۴                | ۴               | غیر اول     | A - B - C - D - F - H         |

❖ رسم گراف و اعداد متفاوت دیگر برای مقدار n به شرطی که مسیرهای آزمون را پوشش داده باشد نیز مورد قبول است.

(اسلایدهای آزمون - فصل ۲۲ و ۲۳ و ۲۴ - اسلایدهای ۵۴ و ۵۵ و ۵۶)

۵- کپل خان را می‌شناسید؟ ایشان دوست قدیمی من (استاد) است. معمولاً در اغلب امتحانات مهندسی نرم افزار حضور دارد. کپل خان است دیگر! این ترم به این فکر افتاده تا به همراه سه نفر از دوستانش یعنی تپل و مپل و خپل یک Startup ایجاد کند. ایده این است، یک فروشگاه آنلاین می‌سازیم که از مزایای شبکه اجتماعی بهره بگیرد؛ شاید هم برعکس، یک شبکه اجتماعی می‌سازیم و در آن امکان فروش را فراهم می‌کنیم. در توضیحات بیشتر اینطور بیان می‌کند که، ما اغلب برای گذراندن وقت به شبکه‌های مجازی سر می‌زنیم. گاهی هم خرید می‌کنیم. به فروشگاه‌های آنلاین برای خرید می‌رویم. گاهی هم امکاناتی مثل لایک و غیره می‌بینیم. حال فرض کنید در یک فروشگاه مجازی، فروشندگان حضور دارند و افراد آنها را فالو می‌کنند و می‌توانند تعامل مستقیم داشته باشند. اگر ما بخواهیم از آنها خرید کنیم به راحتی این کار انجام شده و مراحل خرید مانند پرداخت، حمل و غیره توسط سیستم کپل خان پیگیری می‌شود. اطلاعات و داده‌ها در همین سطح است. بر اساس این توضیحات، سوالاتی که برای کپل خان مطرح است، و انتظار می‌رود شما به آنها پاسخ دهید به شرح زیر می‌باشد:

- A. با توجه به روش‌های استخراج نیازمندی‌ها که در این درس آموخته‌اید، ۸ مورد از نیازمندی‌های این نرم افزار را مشخص کنید.
- B. چه نیازمندی غیرعملکردی در این سیستم وجود دارد؟
- C. برای انجام این پروژه چه متدولوژی را پیشنهاد می‌کنید؟ چرا؟
- D. چه سطوحی از آزمون برای این سیستم مورد نیاز است و چرا؟

۶- با توجه به توضیحات ارائه شده در سوال کپل خان (سوال ۵) به سوال زیر پاسخ دهید.

- A. آیا معماری میکروسرویس را به کپل خان پیشنهاد می‌کنید؟ چرا؟
- B. به نظر شما Decomposition میکروسرویس‌ها باید با چه الگویی انجام شود. هر الگویی که پیشنهاد می‌دهید را کمی توضیح داده و سه نمونه از میکروسرویس‌های پیشنهادی را نام ببرید.
- C. آیا ممکن است برای این سیستم Saga کاربرد پیدا کند؟ مثال بزنید.
- D. چه روش Messaging برای این سیستم پیشنهاد می‌کنید؟ روش انتخابی خود را توضیح دهید.

موفق باشید

تیم آموزش مهندسی نرم افزار