به نام خدا

**پاسخ‌نامه آزمون درس مهندسی نرم‌افزار پیشرفته**

**دانشجو:**

**فاطمه فتحی‌جم**

**استاد راهنما:**

**دکتر علی رضوی ابراهیمی**

خرداد ۱۴۰۰

پاسخ سؤالات میان‌ترم

**1- در معنی و مفهوم علمی function point و ترجمه مناسب آن به فارسی بحث نمایید.**

مفهوم function point از جمله مواردی است که در مبحث متریک‌های پروژه و فرآیند نرم‌افزار مورد بررسی قرار می‌گیرد. بدین منظور ابتدا دررابطه‌با معیارهای اندازه‌گیری پروژه‌های نرم‌افزاری مقدمه‌ای شرح می‌دهیم و سپس functio point را موردبحث و بررسی قرار می‌دهیم.

**متریک و اندازه‌گیری چیست؟** متریک‌های کمیتی هستند که به افرادی که با نرم‌افزار کار می‌کنند امکان می‌دهند تا نسبت به کارآمد بودن فرآیند و پروژه‌های نرم‌افزاری بینش پیدا کنند. همچنین متریک برای مشخص کردن نواحی و حیطه‌های مشکل‌آفرین به کار می‌روند، بطوریکه راه‌حل‌هایی پیدا شود و فرآیند نرم‌افزاری بهبود یابد.

**چرا معیارهای اندازه‌گیری اهمیت دارد؟** اگر شما اندازه‌گیری نکنید، قضاوت شما فقط می‌تواند بر اساس ارزیابی ذهنی باشد اما به‌وسیله اندازه‌گیری می‌توان روند را (چه خوب چه بد) دریافت و برآوردهای بهتری انجام داد و طی زمان، بهبودها و پیشرفت‌های واقعی اتفاق می‌افتند. اندازه‌گیری نرم‌افزار در دنیای فیزیکی به دو طریق اندازه‌های مستقیم و اندازه‌های غیرمستقیم طبقه‌بندی می‌شود.

اندازه‌های مستقیم در فرآیند مهندسی نرم‌افزار شامل هزینه و تلاش (نیروی انسانی) بکار گرفته می‌شوند که عبارت‌اند از: تعداد خطوط کد نوشته شده، سرعت اجرا، اندازه حافظه و معایبی که در یک واحد زمانی گزارش شده‌اند. اندازه‌های غیرمستقیم محصول عبارت‌اند از: عملکرد، کیفیت، پیچیدگی، کارایی، قابلیت اطمینان، قابلیت تعمیر و نگهداری و بسیاری از توانایی‌های دیگر.

**Function Point چیست؟**

با استناد بر کتاب مهندسی نرم‌افزار پیشرفته پرسمن، Function Point یکی از معیارهای اندازه‌گیری مبتنی بر عملکرد (Function-Based Metrics) است. به‌عبارت‌دیگر Function Point عبارت است از معیاری عملکردگرا که به طور گسترده استفاده می‌شود. محاسبه نقطه عملکرد بر اساس ویژگی‌های حوزه اطلاعات و پیچیدگی یک نرم‌افزار است. از معیار نقطه عملکرد (FP) به‌عنوان ابزاری برای اندازه‌گیری عملکرد ارائه شده توسط سیستم **به طور مؤثر** استفاده می‌شود. از جمله موارد استفاده FP می‌توان به موارد زیر اشاره کرد:

* تخمین هزینه یا تلاش لازم برای طراحی، کدنویسی و تست نرم‌افزار
* پیش‌بینی تعداد خطاهایی احتمالی در هنگام تست نرم‌افزار
* پیش‌بینی تعداد اجزاء و تعداد خطوط نوشته شده در سیستم پیاده‌سازی شده

نقاط عملکرد[[1]](#footnote-1) از یک رابطه تجربی بر اساس معیارهای قابل‌اندازه‌گیری و شمارش (مستقیم) دامنه اطلاعات نرم‌افزار و ارزیابی کیفی پیچیدگی نرم‌افزار به دست می‌آیند. پنج ویژگی دامنه اطلاع‌رسانی تعیین شده‌اند که از جمله آن‌ها می‌توان به تعداد ورودی‌های کاربر، تعداد خروجی‌های کاربر، تعداد درخواست‌های کاربر، تعداد فایل‌ها و تعداد رابطه‌های خارجی اشاره کرد.

**2- در معنی و مفهوم رابطه «تضمین کیفیت نرم‌افزار» و «کیفیت نرم‌افزار» بحث نمایید.**

به‌منظور بررسی رابطه بین کیفیت نرم‌افزار و تضمین کیفیت نرم‌افزار ابتدا مفهوم هریک را به طور جداگانه بررسی می‌کنیم.

"دو جنبه متداول از کیفیت وجود دارد: یکی به‌عنوان یک واقعیت عینی مستقل از وجود انسان است. دیگری با آنچه در نتیجه واقعیت عینی فکر می‌کنیم، احساس می‌کنیم یا احساس می‌شود، ارتباط دارد. به‌عبارت‌دیگر، یک کیفیت ذهنی وجود دارد." این اولین تعریفی است که از کیفیت توسط شوهارت ارائه شده است. امروزه اما تعاریف مختلف زیادی از کیفیت وجود دارد. در برخی تعاریف کیفیت "توانایی یک محصول نرم‌افزاری برای مطابقت با الزامات" عنوان شده، اما برخی معتقدند کیفیت مترادف با "ارزشمندی برای مشتری" می‌باشد. به‌طورکلی کیفیت نرم‌افزار در مهندسی نرم‌افزار به دو رده مرتبط و درعین‌حال مجزا بخش‌بندی می‌شود:

کیفیت عملیاتی نرم‌افزار[[2]](#footnote-2): شاخصی جهت نشان دادن میزان تطابق نرم‌افزار با نیازمندی‌های عملیاتی تعریف شده برای نرم‌افزار.

کیفیت ساختاری نرم‌افزار[[3]](#footnote-3): کیفیت ساختاری نرم‌افزار منعکس‌کننده میزان دستیابی به نیازمندی‌های غیرعملیاتی مانند استحکام[[4]](#footnote-4) و قابلیت نگهداری[[5]](#footnote-5) نرم‌افزار است.

**تضمین کیفیت نرم‌افزار چیست و چه رابطه‌ای با کیفیت نرم‌افزار دارد؟**

تضمین کیفیت شامل حسابرسی (وارسی) و گزارش کارکردهای مدیریت است. هدف تضمین کیفیت، فراهم‌کردن اطلاعات لازم برای مدیریت است که به‌وسیله آن‌ها از کیفیت تولید آگاه می‌شود و در نتیجه اطمینان کامل حاصل می‌کند که کیفیت تولید در جهت برآوردن اهداف است. البته اگر اطلاعاتی که از راه تضمین کیفیت حاصل شده‌اند، مشکلاتی را شناسایی کنند. این وظیفه مدیریت است که مشکلات را موردتوجه قرار دهد و منابع لازم برای حل مشکلات مربوط به کیفیت را به کار اندازد.

نکته قابل‌توجه در زمینه تضمین کیفیت نرم‌افزار این‌که تمامی توسعه دهندگان و تولیدکنندگان نرم‌افزار به‌اتفاق بر این باورند هدف اصلی ما از تولید نرم‌افزار، ارائه کیفیتی قابل‌قبول است. اما اینکه چگونه کیفیت را تعریف کنیم نقشی مهمی در روند تضمین کیفیت نرم‌افزار دارد. در بالا کیفیت نرم‌افزار را با تعریف موجود شرح دادیم، اما شاید این جمله بتواند تعریف کیفیت نرم‌افزار را به چالش بکشد. "هر برنامه‌ای دارای چیزهای خوبی دارد، فقط ممکن است آن چیزی نباشد که ما از آن می‌خواهیم".

در زمینه تضمین کیفیت نرم‌افزار، آنچه ضروری است، مسئولیتی است که تمام دست‌اندرکاران ساخت آن از جمله مهندسان نرم‌افزار، مدیران پروژه، مشتریان و تمامی افرادی که در گروه تضمین کیفیت نرم‌افزار یا SQA کار می‌کنند، برعهده داشته و نسبت به آن پایبند هستند.

از جمله فعالیت‌هایی که در قبل تضمین کیفیت نرم‌افزار انجام می‌گیرد می‌توان به موارد زیر اشاره کرد:

* تهیه یک طرح SQA برای پروژه
* شرکت در توسعه توصیف فرایند نرم‌افزاری پروژه
* بازبینی فعالیت‌های مهندسی نرم‌افزار برای وارسی مطابقت با فرایند نرم‌افزاری مشخص
* بازرسی محصولات کاری برای وارسی مطابقت با محصولات تعیین شده به‌عنوان بخشی از فرایند نرم‌افزار
* حصول اطمینان از مستندسازی انحرافات در کار نرم‌افزار و محصولات کاری و مقابله با آنها بر اساس رویه مستندسازی شده
* ثبت هرگونه عدم مطابقت و گزارش به مدیریت ارشد

**طرح SQA چیست؟**

راهنمایی برای نهادینه کردن تضمین کیفیت نرم‌افزار فراهم می‌آورد، ساختار آن بر اساس استاندارد IEEE عبارت است از:

* هدف و دامنه کاربرد طرح
* توصیفی از همه محصولات کاری مهندسی نرم‌افزار
* همة استانداردهای قابل‌استفاده در طول فرآیند نرم‌افزار
* وظایف و کنش‌های SQA ازجمله مرورها و ممیزی‌ها
* ابزارها و روش‌هایی که از وظایف و کنش‌هایی که SQA پشتیبانی می‌کند.
* روش‌های مونتاژ ایمن‌سازی و نگهداری از کلیه سوابق مرتبط با SQA
* نقش‌ها و مسئولیت‌های سازمانی در قبال کیفیت محصول

پاسخ سؤالات پایان‌ترم

**1- چگونه می‌توان در محیط دانشگاه تیم‌های دانشجویی برای تولید «مؤثر» نرم‌افزار تشکیل داد؟**

تولید مؤثر نرم‌افزار در محیط دانشگاه نیازمند انگیزه و یا یک محرک قوی است تا دانشجو را ملزم به مشارکت و فعالیت در زمینه مذکور کند. درگذشته بسیاری از افراد برای آن که بلافاصله پس از گذراندن تحصیلات وارد بازار کار شوند، به سراغ رشته مهندسی کامپیوتر می‌رفتند و در این زمینه نیز موفق بودند اما امروزه می‌بینیم که در آمارهای اعلام شده وزارت کار، بیشترین فارغ‌التحصیلان بیکار، نام فارغ‌التحصیلان رشته‌های کامپیوتری و زیرشاخه‌های آن نیز به چشم می‌خورد. با وجود این آمار نگران‌کننده، متقاضیان ورود به این رشته همچنان نرخی صعودی دارد و دانشگاهیان این حوزه روزبه‌روز بیشتر می‌شوند. به نظر می‌رسد فرآیند تبدیل دانش‌آموز به دانشجو به‌درستی انجام نمی‌گیرد، یعنی دانشجویان دانش لازم و کافی برای ورود به بازار کار را در دانشگاه کسب نمی‌کنند و شاید یکی از مهم‌ترین دلایل این امر، عدم تشکیل تیم‌های دانشجویی و تولید نرم‌افزار در محیط دانشگاه باشد. یک رویکرد برای حل این مشکل می‌تواند ایجاد انگیزه در دانشجویان باشد، یعنی بنابر قوانین و پروتکل‌هایی به دانشجویان این وعده داده شود که در صورت مشارکت در تیم‌های دانشجویی و تولید نرم‌افزار، اقداماتی جهت به‌کارگیری آن‌ها در محیط‌های واقعی کار، انجام می‌گیرد. اما موضوع مهم در اینجا آن است که باید نرم‌افزاری که در دانشگاه توسط دانشجویان تولید می‌شود استانداردهای لازم را جهت ترغیب دانشجویان داشته باشد. در واقع از حداقل اصول کلی در مهندسی نرم‌افزار برخوردار باشد.

اصول کلی که دیوید هوکر در مهندسی نرم‌افزار به‌عنوان یک کلیت است نام برد عبارت‌اند از: دلیل وجود سیستم، ساده نگه‌داشتن، حفظ چشم‌انداز، **آنچه که شما تولید می‌کنید دیگران مصرف می‌کنند،** آینده‌نگری، برنامه‌ریزی پیشاپیش برای استفاده مجدد، تفکر.

شاید مهم‌ترین اصل در موفقیت تولید پروژه مؤثر در تیم‌های دانشجویی، اصل چهارم یعنی "آنچه که شما تولید می‌کنید دیگران مصرف می‌کنند" باشد. چراکه زمانی که دانشجو بداند زمانی که برای یادگیری، طراحی و تولید یک نرم‌افزار صرف می‌کند، قرار است به دست مشتریان برسد و در دنیای واقعی مورداستفاده قرار گیرد، انگیزه بیشتری برای شرکت در این‌گونه فعالیت‌ها خواهد داشت.

**2- نقش «مهندسی نرم‌افزار پیشرفته» برای تأثیرگذاری در تحقق شعار سال «تولید، پشتیبانی‌ها و مانع‌زدایی‌ها» چه می‌تواند باشد؟**

همه‌ساله رهبر معظم انقلاب، متناسب با شرایط کلی حاکم بر کشور سال را به نامی مزین می‌کنند تا این مهم بتواند در سرلوحه برنامه‌ریزی‌های مسئولان و مردم قرار بگیرد. ایشان در نوروز سال ۱۴۰۰ نیز، این سال را به نام سال تولید، پشتیبانی‌ها و مانع زدایی‌ها نام‌گذاری کردند. در این بخش قصد داریم نقش درس مهندسی نرم‌افزار پیشرفته را برای تأثیرگذاری در تحقق شعار سال بررسی نماییم.

پیش از هر چیز بهتر است کمی دررابطه‌با معنا و مفهوم تولید که کلمه کلیدی شعار سال است، صحبت کنیم. تولید یا فرآوری، از اصطلاحات علم اقتصاد، به معنی تهیه کالا و خدمات موردنیاز با استفاده از منابع و امکانات موجود است. تولید بر دو نوع است، تولید مداوم و تولید متناوب و فعالیّت تولیدی سلسله اقداماتی است که برای تبدیل منابع به کالاهای موردنیاز صورت می‌گیرد. نکته بسیار مهم اینجاست که نیاز به تولید، **منجر به پیشرفت تمام رشته‌های مهندسی** شده است. در مهندسی نرم‌افزار نیز تولید نرم‌افزار مؤثر و کارآمد می‌تواند تأثیر بسزایی در تحقق شعار سال داشته باشد.

درس مهندسی نرم‌افزار پیشرفته نیز به طور مفصل به چگونگی تولید نرم‌افزار کارآمد و مهندسی شده می‌پردازد. در درس مهندسی نرم‌افزار ما می‌فهمیم که کار مهندس نرم‌افزار چیست]تولید نرم افزار[ و مهندسی نرم افزار چه اهدافی را دنبال می‌کند]فرآیندی است که عملکرد کارایی و اجرای مطلوب را فراهم می ساز[. بنابراین مهندس نرم افزار می‌تواند با تولید نرم افزار کارآمد در جهت مانع زدایی ها قدم برداشته و به تحقق شعار کمک نماید.

**3- تجزیه و تحلیل نمائید مشکل اجزای نظام آموزشی از ابعاد مختلف چیست که دانشجویان دوره‌های کارشناسی و کارشناسی ارشد در این حوزه قادر به تولید نرم‌افزار و ارائه خروجی مطلوب نمی‌باشند. سپس، برای حل مشکل کارآمدی دانشجویان مهندسی نرم‌افزار برای مشارکت در تولید نرم‌افزار در دوران دانشجویی و یا پس از آن پیشنهاداتی را ارائه نمایید؟ (راهنمایی: تلاش نمایید بعضی از فرایندهای نرم‌افزار موردمطالعه در درس را با اصلاحاتی جزئی برای کارآمدی در محیط دانشگاه مناسب‌سازی نمایید).**

دررابطه‌با بخش اول سؤال، مشکل را می‌توان از جنبه‌های مختلف بررسی کرد اما پایه و اساس مشکل در نحوه پذیرش دانشگاه‌هاست. بی‌شک ورودی‌های دانشگاه هستند که خروجی آن را می‌سازند. موردی که شاید خیلی موردبحث واقع نشود اما نباید از مشکلات و تنگناهای ورود به دانشگاه و پدیده‌ای به نام کنکور غافل بود، چراکه امروزه دانش‌آموزان بیشتر بر اساس مطلوبیت عمومی رشته‌ها تصمیم‌گیری می‌کنند و این امر سبب شده است تا گروه زیادی از آنها در رشته‌هایی ادامه تحصیل دهند که ممکن است متناسب با توانمندی‌های آنان نباشد و مسلماً خروجی این فرایند گروهی از افرادی است که دارای تحصیلات عالیه‌اند **اما از کارآمدی چندانی برخوردار نیستند**. اما اگر ورودی دانشگاه بر اساس **توانمندی** و علاقه‌مندی افراد باشد خروجی آن نیز نسلی خواهد بود که متناسب با توانمندی‌ها و استعدادهای واقعی خود آموزش‌دیده و آماده ورود به عرصه کار و تولید هستند.

اما فرض می‌کنیم دانشجو متناسب با علاقه و توانمندی‌های خود رشته مهندسی نرم‌افزار را برگزیده و از حداقل قابلیت‌ها برای موفقیت در این رشته برخوردار است. اما همچنان خروجی مطلوبی ارائه نمی‌شود. در این شرایط نیز مشکل از دو منظر قابل‌بررسی است، یکی از دیدگاه دانشجو و دیگری از دیدگاه دانشگاه.

خروجی دانشگاه ارتباط مستقیمی با انگیزه ورود دانشجویان به دانشگاه دارد، اگر انگیزه دانشجو یادگیری دانش مربوط به نرم‌افزار و بهره‌برداری از آن در محیط کار باشد، قطعاً از توانمندی‌های خود استفاده خواهد کرد، به‌عنوان مهندس نرم‌افزار به تولید نرم‌افزار و یا سایر بخش‌های مربوط به تولید نرم‌افزار پرداخته و در نتیجه دانشگاه خروجی مطلوب‌تری از دانشجویان کارشناسی و یا کارشناسی ارشد نرم‌افزار خواهد داشت.

اما نقش دانشگاه نیز در این زمینه بسیار مهم و اساسی است. در واقع دانشگاه می‌تواند با رویکردهای اساسی دانشجوی مهندسی نرم‌افزار را به سمت **تولید نرم‌افزار کارآمد** سوق دهد. یک روش که پیش‌تر نیز گفتیم، مشارکت دادن دانشجوها در پروژه‌های واقعی و کاربردی است. فرآیند تولید نرم‌افزار کارآمد توسط دانشجوها در محیط دانشگاه و یا پس از آن باید با اصول مهندسی نرم‌افزار و فرآیند تولید نرم‌افزار مطابقت داشته باشد، بدین منظور در ادامه فرآیند تولید نرم‌افزار را مرور می‌کنیم.

مفهوم فرآیند نرم‌افزار: مجموعه‌ای از فعالیت‌ها تحت یک چارچوب خاص و همچنین وظایف مهندس نرم‌افزار برای به انجام رساندن کار. فرآیند نرم‌افزار تعریف، اجرا و پیاده‌سازی، ارزیابی، اندازه‌گیری، مدیریت، ایجاد تغییر و بهبود فرایند چرخه حیات خود نرم‌افزار از جمله مراحلی است که فرآیند نرم‌افزار را تشکیل می‌دهند.

**برنامه‌ریزی و تجزیه و تحلیل نیازمندی‌ها**

تجزیه و تحلیل نیازمندی‌ها، مهم‌ترین و اساسی‌ترین مرحله در روند تولید نرم‌افزار محسوب می‌گردد. این امر توسط کارشناسان ارشد و اعضای متبحر تیم با استناد به داده‌های ورودی مشتریان، بخش فروش و بررسی‌های بازار و متخصصین این حوزه از صنعت انجام می‌گیرد. نتیجه مطالعات و امکان‌سنجی‌ها موجب به‌دست‌آوردن دیدگاه‌های مختلف فنی خواهد بود که می‌تواند باعث پیاده‌سازی موفقیت‌آمیز پروژه با حداقل ریسک گردد.

تعریف نیازمندی‌ها

زمانی که تجزیه و تحلیل‌ها به پایان می‌رسد مرحله بعد یعنی تعریف دقیق نیازمندی‌ها و مستند نمودن آن آغاز می‌شود. این مستندات شامل نیازهای نرم‌افزار موردنظر بوده که مورد تأیید مشتری و همچنین تحلیل گران بازار قرار گرفته است.

طراحی معماری نرم‌افزار

در طراحی نرم‌افزار کلیه و تمامی جنبه‌های معماری و همچنین روش‌های ارتباطی و جریان ارسال و دریافت داده‌ها از سامانه‌های دیگر مدنظر قرار می‌گیرد. در معماری پیشنهادی، می باید طراحی داخلی کلیه مدل‌ها و پیمانه‌ها به نحوی کاملاً واضح و آشکار و با ذکر تمامی جزئیات در سند مشخصات طرح درج گردد.

تولید نرم‌افزار

در این مرحله از تولید نرم‌افزار، روند عملیاتی آغاز و نرم‌افزار تولید می‌گردد. در این مرحله کد برنامه‌ریزی به همان صورتی که در سند مشخصات طرح تأکید گردیده تهیه می‌گردد. چنانچه طرح بر مبنای جزئیات مندرج در سند مذکور و به‌صورت سازمان داده شده پیاده شود تولید نیز بدون هیچ مشکلی انجام خواهد شد تولیدکنندگان نرم‌افزار می باید از رهنمودهای کدگذاری که به‌وسیله سازمان خود و ابزار برنامه‌نویسی تعریف شده است تبعیت نمایند.

آزمایش محصول

این مرحله معمولاً زیرمجموعه‌ای از کلیه مراحل دیگر در مدل‌های نوین چرخه حیات تولید نرم‌افزار محسوب می‌گردد. عملیات مربوط به آزمایش محصول اغلب در کلیه مراحل تولید محصول انجام می‌گیرد. ولی آزمایش در این مرحله زمانی انجام می‌گیرد که نقص یا کاستی‌هایی مشاهده و یا گزارش شده است. بدین لحاظ کارشناسان پس از پیگیری نقص و یافتن علل، نقص یا کاستی را برطرف و مجدداً تا زمانی که محصول در سطح استانداردی که در سند مشخصات نیازمندی‌های نرم‌افزار مشخص گردیده است قرار گیرد، نرم‌افزار را آزمایش می‌نمایند.

پیاده‌سازی در بازار و نگهداری

زمانی که نرم‌افزار مراحل نهائی آزمایش را طی و آماده پیاده‌سازی شد، عرصه رسمی آن در یک بازار ویژه آغاز می‌گردد. گاهی اوقات پیاده‌سازی نرم‌افزار در یک سازمان به‌صورت مرحله‌ای انجام می‌گیرد و این امر بستگی به استراتژی کسب‌وکار سازمان مذکور دارد.

درصورتی‌که دانشجویان مهندسی نرم‌افزار این امکان را داشته باشند که در تمامی مراحل فوق‌الذکر حضور داشته و با استانداردهای تولید و مهندسی نرم‌افزار آشنا شوند، قطعاً خروجی مطلوب‌تری از دانشگاه‌ها خواهیم داشت.

1. Function Points [↑](#footnote-ref-1)
2. Software Functional Quality [↑](#footnote-ref-2)
3. Software Structural Quality [↑](#footnote-ref-3)
4. Robustness [↑](#footnote-ref-4)
5. Maintainability [↑](#footnote-ref-5)