

به نام خدا

۹۹۳۱۰۹۸

ابراهیم صدیقی

سوال (۱)

fault: یک fault نقصی در کد است که به طور بالقوه می تواند باعث fault شود. این اشتباهی است که توسعه دهنده در مرحله اجرا مرتکب شده است. به عنوان مثال، یک تابع ساده را در نظر بگیرید که مربع یک عدد را در یک برنامه ماشین حساب محاسبه می کند. اگر توسعه دهنده به اشتباه کد را بنویسد تا به جای مربع کردن عدد در خودش ضرب کند، این یک fault در کد است.

error: یک error زمانی رخ می دهد که یک fault در شرایط خاص اجرا شود و منجر به انحراف قابل مشاهده از رفتار مورد انتظار می شود. با توجه به مثال قبلی، اگر کاربر برای محاسبه مربع خود عدد 3 را در ماشین حساب وارد کند اما به جای 3^2 (9) عدد $3*3$ (9) را دریافت کند، این اختلاف خطای ناشی از error در کد.

failure: failure زمانی اتفاق می افتد که یک error خود را نشان می دهد و با ایجاد نتایج یا رفتار سیستم نادرست، بر تجربه کاربر نهایی تأثیر می گذارد. در ادامه سناریوی ماشین حساب، اگر محاسبه مربع نادرست ادامه یابد و کاربران به طور مداوم نتایج نادرست را هنگام تربیع اعداد دریافت کنند، این یک failure نرم افزار محسوب می شود.

سوال ۲) رابطه بین fault، failure، testing و debugging پیچیده و به هم مرتبط است. test برای کشف fault قبل از اینکه منجر به failure در یک محیط زنده بشود ضروری است. هنگامی که failure رخ می دهد، debugging برای شناسایی و اصلاح fault ایجاد کننده failure ضروری است. با انجام test کامل و debug موثر، توسعه دهندگان می توانند کیفیت و قابلیت اطمینان نرم افزار را افزایش دهند.

سوال ۳)

(2

الف) برای ورودی با آرایه خالی خط $arr(len / total = average)$ تقسیم بر صفر خواهد داشت و برای ورودی آرایه None در خط : `arr` for به تقسیم می خوریم. البته فرض شده که ورودی تابع آرایه است، اگر ورودی تابع داده ی نوعی دیگری باشد به ارور میخوریم.

ب) همه تست کیس ها فالت را اجرا میکنند.

پ) هر آرایه غیر خالی یا غیر None مثال [1,2,3]

ت) غیر ممکن

ث) [] یا None

(4

الف) کد از نظر منطقی درست کار میکند و با آرایه خالی هم میتواند کار کند اما برای None مشکل دارد که در خط `arr(len = n)` مشکل ایجاد میکند.

ب) نا ممکن

پ) هر تست کیسی غیر از None مثال [1,2,3]

ت) غیر ممکن، تنها تست کیسی که منجر به ارور میشود قطعا منجر به شکست هم میشود.

ث) None