

بسمه تعالی



تمرین دوم درس اصول طراحی کامپیوتر

طراح: پارسا کنگاوری

استاد: جناب آقای دکتر پارسا

## سوالات تئوری

### سوال اول:

گرامر مستقل از متن زیر را در نظر بگیرید:

$$S \rightarrow SS+ \mid SS^* \mid a$$

نشان دهید که چگونه رشته  $aa+a^*$  توسط این گرامر تولید میشود. Parse tree مربوط به این رشته را رسم کنید.

### سوال دوم:

گرامر  $LL(1)$  زیر را در نظر بگیرید. با استفاده از توابع  $first$ ,  $follow$ ، جدول تجزیه این گرامر را رسم کنید.

$$E \rightarrow TE'$$

$$E' \rightarrow +TE' \mid \epsilon$$

$$T \rightarrow FT'$$

$$T' \rightarrow *FT' \mid \epsilon$$

$$F \rightarrow id \mid ( E )$$

### سوال سوم:

گرامر مستقل از متن  $G$  به شرح زیر مفروض است. تعیین کنید چند زوج قاعده ی تولید در این گرامر با شرایط  $LL(2)$  بودن سازگار نیست؟

$$G: \quad S \rightarrow ASB$$

$$S \rightarrow \lambda$$

$$A \rightarrow aA$$

$$A \rightarrow \lambda$$

$$B \rightarrow bB$$

$$B \rightarrow \lambda$$

## سوالات عملی

### سوال چهارم

یکی از قواعد Clean Code، کامنت گذاری صحیح است. تا حد امکان باید کد خوانا و قابل فهم باشد و نیازی به توضیحات در کامنت‌ها نباشد. بنابراین از حداقل کامنت استفاده کرد. در این تمرین قصد داریم کامنت‌های اضافی و نامنظم را از کد جاوا حذف کنیم. با استفاده از ابزار انتلر یک برنامه جهت refactor کد بنویسید که اصول زیر را در کد اعمال کند:

1. کامنت‌های نوع multiline که تنها در یک خط نوشته شده اند را به کامنت inline تبدیل کند.
2. کامنت‌هایی که در چند خط پشت سر هم آمده باشد (inline و multiline) را تنها خط اولش را باقی بگذارد و سایر خطوط را پاک کند و اگر multiline بود به inline تبدیل کند.

### سوال پنجم

برای اعمال شرط در کد جاوا یا می‌توانیم از if statement یا inline statement استفاده کرد. میدانیم استفاده از inline statement، کارایی بالاتری دارد اما از دیدگاه clean code اصلاً مناسب نیست. چرا که خوانایی کد را بسیار تحت تاثیر قرار میدهد. در این تمرین قصد داریم که یک کد جاوا را در ورودی گرفته و تمامی inline statement ها را به if statement تبدیل کند.

😊 موفق باشید