آزمون میانی درس طراحی کامپایلر

مجموع نمره‌ها: ۱۰۰ زمان آزمون: ۹۰ دقیقه

---

۱) به این سؤال‌ها پاسخ دهید.

یک) استفاده از کد میانی چگونه به توسعه‌ی یک کامپایلر (افزودن زبان‌های متفاوت، تولید کد برای معماری‌های متفاوت، پیاده‌سازی بهینه‌سازی‌ها) کمک می‌کند؟

دو) برای برخی از زبان‌های برنامه‌نویسی هم مفسر[۱] و هم مترجم[۲] وجود دارد. از دید اجرای یک برنامه‌ی ورودی چه تفاوتی بین آنها وجود دارد؟ چه مزیتی مترجم‌ها نسبت به مفسرها دارند؟

۲) اعداد ممیز شناور[۳] دودویی[۴] (یعنی فقط از اعداد صفر و یک تشکیل شده باشند) را در نظر بگیرید: این اعداد می‌توانند علامت‌دار باشند و از نمایش علمی (که با حرف بزرگ E و سپس توان مشخص می‌شود) نیز استفاده نمایند که در این صورت توان (عدد قرار گرفته بعد از E) نیز دودویی است و می‌تواند علامت‌دار («+» یا «-») باشد. چند مثال از این اعداد: «1»، «1.10E10»، «+1.0E-100»، «-11E+11». دقت کنید که در این اعداد، ممیز (نقطه) حداکثر یک بار در قسمت عدد ظاهر می‌گردد و قبل و بعد از آن حتما یک عدد قرار می‌گیرد.

یک) برای این اعداد یک عبارت منظم ارائه دهید.

دو) برای شناسایی این عبارت منظم یک NFA بسازید.

سه) NFA ساخته شده در قسمت یک را به یک DFA تبدیل نمایید.

چهار) با ترکیب State-های DFA بخش سوم، DFA کمینه‌ی آن را بسازید.

---

۱ Interpreter
۲ Compiler
۳ Floating point
۴ Binary

۳) گرامر زیر با الفبای $\{a, b, c\}$ را در نظر بگیرید.

```
S → SAB
S → SBB
S → c
A → a
B → b
```

یک) عمل Left factoring و حذف Left recursion را روی این گرامر انجام دهید.

دو) توابع FIRST و FOLLOW را برای Nonterminal-ها محاسبه نمایید.

سه) جدول LL(1) را برای این گرامر تشکیل دهید. آیا این گرامر LL(1) است؟

۴) گرامر زیر با الفبای $\{*, (,), n\}$ را در نظر بگیرید.

```
S → * S
S → S ( S )
S → A
A → ( S )
A → n
```

یک) با ارائه‌ی درخت تجزیه[۱] و چپ‌ترین اشتقاق[۲] برای رشته‌ی «n((n))*»، نشان دهید این گرامر مبهم است.

دو) مجموعه‌ی Itemset-های الگوریتم LALR(1) را برای این گرامر محاسبه کنید.

سه) جدول ACTION و GOTO را برای الگوریتم LALR(1) بسازید. آیا این گرامر LALR(1) است؟

چهار) جدول قسمت سه را به صورتی تغییر دهید که اولویت قاعده‌ی اول گرامر از قاعده‌ی دوم بیشتر باشد. سپس رشته‌ی قسمت یک این سؤال را با استفاده از جدول LALR(1) و نمایش وضعیت پشته و ورودی تجزیه نمایید.

نمره‌ی سؤال‌ها

سؤال ۱:	قسمت یک	۵	قسمت دو	۵			
سؤال ۲:	قسمت یک	۱۰	قسمت دو	۵	قسمت سه	۵	قسمت چهار ۵
سؤال ۳:	قسمت یک	۵	قسمت دو	۵	قسمت سه	۱۵	
سؤال ۴:	قسمت یک	۵	قسمت دو	۱۰	قسمت سه	۱۵	قسمت چهار ۱۰

۱ Parse Tree

۲ Left-most derivation