

## پروژه اول درس نظریه زبان‌ها و ماشین‌ها

### لطفاً به نکات زیر توجه فرمایید:

- گزارش باید به زبان فارسی در قالب فایل WORD و PDF به همراه پوشه‌ای به نام code که همه در یک فایل فشرده شده‌است، باشند.
- فایل گزارش را به فرمت P1\_report\_StdNum.pdf نام‌گذاری نمایید (همانند (P1\_report\_97131.(pdf|doc) و توجه داشته باشید که ارسال تمرین بدون گزارش فاقد ارزش است.
- برای هر سوال جداگانه باید فایل کد، با یکی از زبان‌های Python، Java و C++ نوشته شود. کامنت گذاری در حد لازم نیز انجام پذیرد. فرمت نامگذاری فایل اصلی مربوط به هر بخش از تمرین متناسب با فرمت P1\_ProblemNum\_StdNum و در پوشه Code ذخیره شده باشد.
- فایل‌های کد و گزارش خود را مطابق فرمت‌های فوق آماده و در قالب یک فایل فشرده با نام P1\_StdNum.zip تهیه نمایید.
- مهلت ارسال پروژه یک ۱۴۰۲/۱۶/۰۳ می‌باشد. در ضمن تاخیر در ارسال پروژه مشمول کسر نمره خواهد شد.
- مطابق قوانین دانشگاه هر نوع کپی برداری، اشتراک کار دانشجویان و استفاده مستقیم از کدهای اینترنت غیرمجاز است.
- هرگونه سوال و مشکل در خصوص صورت پروژه را از طریق آدرس [r.ameri@aut.ac.ir](mailto:r.ameri@aut.ac.ir) پیگیری نمایید.

۱. سیستمی را طراحی و پیاده سازی نمایید که مشخصات پذیرنده متناهی قطعی<sup>۱</sup> (DFA) را از فایل ورودی دریافت کرده می‌کند و بررسی کند که رشته‌های وارد شده توسط کاربر در DFA پذیرفته می‌شود یا پذیرفته نمی‌شود. شایان ذکر است باید رشته را از ورودی برنامه و همچنین پذیرنده را در قالب فایل DFA\_Input\_1.txt دریافت کند که ساختار این فایل در توضیحات پیاده سازی ذکر شده است. در ضمن به عنوان نمونه DFA، فایلی به همین نام در پوشه این پروژه موجود می‌باشد.

<sup>۱</sup> Deterministic Finite Acceptor

## پروژه اول درس نظریه زبان‌ها و ماشین‌ها

۲. سیستمی را طراحی و پیاده سازی نمایید که پذیرنده متناهی غیر قطعی<sup>۲</sup> (NFA) را به پذیرنده متناهی قطعی (DFA) تبدیل نماید. NFA ورودی را در قالب فایل NFA\_Input\_۲.txt دریافت نموده و DFA خروجی را نیز در قالب فایل DFA\_Output\_۲.txt تولید نماید که ساختار این فایل‌ها در توضیحات پیاده سازی ذکر شده است. در ضمن به عنوان نمونه NFA، فایل به نام NFA\_Input\_۲.txt در پوشه این پروژه موجود می‌باشد.

۳. سیستمی را طراحی و پیاده سازی نمایید که عبارت منظم<sup>۳</sup> را به پذیرنده متناهی غیر قطعی (NFA) تبدیل نماید. عبارت منظم ورودی را در قالب فایل RE\_Input\_۳.txt دریافت نموده و NFA خروجی را در قالب فایل NFA\_Output\_۳.txt تولید نماید که ساختار این فایل‌ها در توضیحات پیاده سازی ذکر شده است. در ضمن به عنوان نمونه عبارت منظم، فایل به نام RE\_Input\_۳.txt در پوشه این پروژه موجود می‌باشد.

قالب ورودی برای عبارت منظم بدین صورت می‌باشد:

- ✓ در خط اول حروف الفبا قرار دارند که با فاصله و یا space از هم جدا شده‌اند.
- ✓ در خط دوم عبارت منظم مورد نظر نوشته شده است و از <sup>^</sup> برای توان استفاده می‌کنیم

نمونه یک عبارت منظم:

a b  
 $(a+b)^a * b$

### توضیحات پیاده‌سازی

۱. نیازی نیست که DFA و NFA حاصل بهینه باشد. می‌دانیم که برای هر زبان مشخص می‌توان چندین DFA و NFA داشت.
۲. قالب ورودی و خروجی برای DFA و NFA بدین صورت می‌باشد:
  - ✓ در خط اول حروف الفبا قرار دارند که با فاصله و یا space از هم جدا شده‌اند.
  - ✓ در خط دوم حالات ماشین آمده است که با فاصله و یا space از هم جدا شده‌اند.
  - ✓ در خط سوم نام حالت شروع قرار دارد.

<sup>۲</sup> Nondeterministic Finite Acceptor

<sup>۳</sup> Regular Expressions

## پروژه اول درس نظریه زبان‌ها و ماشین‌ها

- 
- ✓ در خط چهارم حالات نهایی که با فاصله و یا space از هم جدا شده‌اند، آمده است.
- ✓ از خط چهارم به بعد مقادیر تابع انتقال و یا هر یال از گراف ماشین به صورت “ $q_i a q_j$ ” آمده است به این معنی که در این ماشین یک یال از حالت  $q_i$  با برچسب  $a$  به حالت  $q_j$  وجود دارد.

نمونه یک DFA:

a b  
 $Q_0 \rightarrow Q_1 \rightarrow Q_2$   
 $Q_0$   
 $Q_1$   
 $Q_0 \xrightarrow{a} Q_1$   
 $Q_0 \xrightarrow{b} Q_1$   
 $Q_1 \xrightarrow{a} Q_2$   
 $Q_1 \xrightarrow{b} Q_2$   
 $Q_2 \xrightarrow{a} Q_2$   
 $Q_2 \xrightarrow{b} Q_2$