

فاز چهارم پروژه کامپایلر - اعلامیه‌ی دوم
نیم‌سال اول 86-87 - دکتر ابوالحسنی

موعد تحویل پروژه (کد + مستندات)¹ **۱۲ ظهر روز شنبه ۲۰ بهمن** است. یک‌شنبه همین ساعت نمره‌های این فاز و نمرات نهایی فازهای قبل اعلام می‌شود. ضرایب فازها به ترتیب ۱۰، ۲۰، ۳۰ و ۴۰ درصد است. حداکثر تا دوازده شب یک‌شنبه اعتراضات خود را بفرستید. دوشنبه ظهر نمرات نهایی اعلام می‌شود، و دوازده شب برای استاد ارسال می‌شود. با توجه به فشردگی بودن زمان‌بندی، **در این فاز به پروژه‌های که بعد از موعد فرستاده شود، نمره‌ای تعلق نمی‌گیرد.** در نتیجه حتماً قبل از ارسال چک کنید که پروژه‌تان مشکل کامپایل نداشته‌باشد، و چند بار آن را تست کنید که درست جواب بدهد.

اگر پروژه‌ی خود را زود بفرستید، بررسی خواهد شد، نمره‌ی کارکردش اعلام می‌شود و اگر مستندات آن اشکالی داشت نیز گفته می‌شود. این کار برای هر گروه حداکثر یک بار انجام می‌شود. سپس می‌توانید مجدداً ارسال کنید. آخرین پروژه‌ای که قبل از موعد ارسال بشود مورد ارزیابی نهایی قرار می‌گیرد. هر گونه بهینه‌سازی که در حجم کد خروجی، یا زمان اجرای برنامه‌ی خروجی (به فرض این‌که روی یک ماشین واقعی اجرا می‌شود)، یا در حجم کد کامپایلر، یا زمان کامپایل انجام دهید، در صورت باارزش بودن (که سلیقه‌ی شخصی است) نمره‌ی اضافی دارد. دقت کنید که صحت برنامه‌ی تولیدی خیلی مهم‌تر از انجام بهینه‌سازی است. به مورد ۶ از موارد مستندات برای توضیح بیشتر رجوع کنید. اگر اشکالی دارید یا فکر می‌کنید نیاز به کلاس رفع اشکال هست به abbas.mehrabian@gmail.com ای-میل بزنید.

خواهش می‌شود به این موارد کاملاً دقت کنید: خروجی برنامه در حالتی که ورودی مشکل `syntax` یا `lexical` دارد باید `"syntax error"` باشد. اگر برنامه مشکل `type` دارد، خروجی باید `"semantic error"` باشد. به بزرگی و کوچکی حروف و فاصله‌ها دقت کنید. خروجی را باید در `System.out` بنویسید. ورودی را نیز از `System.in` بخوانید. اگر برنامه درست باشد، باید یک فایل به نام `program.asm` تولید کنید که کد اسمبلی برنامه در آن نوشته شده‌است. در ضمن شبیه‌ساز اسمبلی به بزرگی و کوچکی حروف حساس است.

در بخش `resources` وبسایت درس، سه پروژه‌ی صحیح که تا فاز سوم نوشته شده‌اند، به همراه مستندات آن‌ها قرار داده شده‌است. می‌توانید از هر یک از آن‌ها استفاده کنید، و در این صورت حتماً در مستندات ذکر کنید که از کدام استفاده کرده‌اید. تعدادی برنامه‌ی `MiniJava` نیز قرار گرفته‌است، که

¹ documentation

می‌توانید با کمک آن‌ها کامپایلرتان را تست کنید. برای فهمیدن خروجی درست هر یک از آن‌ها، کافی است آن را با Java کامپایل و اجرا کنید.

در مورد مستندات:

علاوه بر نحوه‌ی کلی انجام کار، موارد زیر را حتماً باید ذکر کنید:

1. چگونه به متغیرهای محلی توابع دسترسی دارید؟ از چه ساختاری استفاده می‌کنید؟
 2. کلاس‌ها را چگونه پیاده‌سازی کرده‌اید؟ به فیلدها و متدها چگونه دسترسی دارید؟
 3. در مورد label هایی که استفاده می‌کنید و نام‌گذاری آن‌ها توضیح دهید.
 4. حلقه‌ی while را چگونه پیاده‌سازی کرده‌اید؟
 5. مقدار عبارات (که ممکن است پیچیده و تودرتو باشند) را چگونه پیدا می‌کنید؟ به طور خاص، درباره‌ی پیاده‌سازی عملگرهای “+” و “.method(args)” و “.length” توضیح دهید.
 6. اگر بهینه‌سازی انجام داده‌اید (برای گرفتن نمره‌ی اضافی)، باید در مورد آن کاملاً توضیح دهید. آیا این بهینه‌سازی به قیمت بدتر شدن قسمت دیگری انجام می‌شود؟ در کدام قسمت کد شما می‌توان انجام شدن این بهینه‌سازی را مشاهده کرد؟ دقیقاً مشخص کنید.
- سعی کنید تا حد امکان دقیق بنویسید و از مجرد نوشتن بپرهیزید. مثلاً به جای این که بگویید فلان چیز را نگه می‌داریم، بگویید فلان چیز را در رجیستر R1 نگه می‌داریم.

در مورد Simulator.java:

لطفاً خط ۲۵۴ این فایل را کامنت کنید.