

توجه: تمرین شامل ۸ صفحه و ۷ پرسش است.

بخش اول: خطاهای برنامه نویسی

پرسش ۱ مباحث مربوط به انواع خطاها در هنگام کامپایل و اجرای برنامه را پوشش می دهد.

۱. انواع مختلفی از خطا توسط کامپایلرها در زمان کامپایل گزارش می شوند. همچنین در صورت کامپایل صحیح یک برنامه و تولید کداجرایی آن در زمان اجرا نیز برنامه ممکن است دچار خطا شود. با توجه به توضیحات مطرح شده در کلاس حل تمرین، خطاهای موجود در برنامه ۱، در زبان ++C استاندارد، را به همراه نوع آن مشخص کنید. توضیح دهید هر خطا چگونه تشخیص داده می شود.

```
1 include <iostream>
2 void main() {
3     const float f = 10.5; const int i = 20
4     int j = 20;
5     int A[j], B[i];
6     B[30] = 15;
7     B[j] = i;
8     B[0] = 1.5;
9     B[1] = i;
10    A[x] = 15;
11    A[19.5] = 15;
12    B[f] = -1;
13    int re@ = 2**3 + 1;
14    j = f + i;
15    return 0;
16 } // end of main function
```

برنامه ۱: تابع main نوشته شده به زبان ++C

بخش دوم: گرامرهای مستقل از متن

پرسش‌های ۲ تا ۶ مباحث مربوط به گرامرهای مستقل از متن و نیز مسئله ابهام در این گرامرها، مشکلات ناشی از آن و چگونگی برطرف کردن این مشکلات را پوشش می‌دهند.

۲. در نظر بگیرید:

i. L_{CS} : مجموعه زبان‌های حساس به متن

ii. L_{CF} : مجموعه زبان‌های مستقل از متن

iii. L_{DCF} : مجموعه زبان‌های مستقل از متن معین (قطعی)^۱

iv. L_{Amb} : مجموعه زبان‌های مبهم (گنگ)^۲

v. L_{Reg} : مجموعه زبان‌های منظم^۳

(آ) رابطه بین قلمرو این مجموعه زبان‌ها را در قالب نمودار ون^۴ رسم کنید.

(ب) برای هریک از نواحی موجود در نمودار ون قسمت (آ) یک زبان (همراه با گرامر) مثال بزنید.

¹Deterministic

²Ambiguous

³Regular

⁴Venn diagram

۳. گرامر مستقل از متن زیر با غیر پایانه S داده شده است:

$$S \rightarrow aSbS \mid bSaS \mid \epsilon$$

(آ) نوع گرامر و وضعیت ابهام آن را مشخص کنید.

(ب) زبان منتج از گرامر، نوع زبان و وضعیت ابهام آن را مشخص کنید.

(ج) جایگاه زبان منتج از این گرامر را در نمودار ون رسم شده برای پرسش ۲.(آ)، تعیین کنید.

۴. گرامر زیر برای دستور `if` در زبان برنامه‌نویسی پاسکال، نوشته شده است (غیرپایانه‌ها با حروف بزرگ شروع شده‌اند و غیرپایانه `Other` به هر دستوری غیر از `if` اشاره می‌کند).

`St` → `if Expr then St`

| `if Expr then St else St`

| `Other`

(آ) توضیح دهید چرا این گرامر برای همه دستورات شرطی مناسب نیست؟

(ب) گرامر را جوری بازنویسی کنید که قابل استفاده در تحلیل نحوی پایین به بالا باشد.

۵. گرامر زیر برای توصیف صوری عبارتهای ریاضی در زبانهای برنامه‌نویسی داده شده است (غیرپایانه‌ها با حروف بزرگ شروع شده‌اند):

$$E \rightarrow E + E$$

$$| E - E$$

$$| E * E$$

$$| E / E$$

$$| id | no | (E)$$

این گرامر مبهم است. می‌توان ابهام در این گرامر را با در نظر گرفتن اولویت (تقدم و شرکت‌پذیری) برای عملگرها از بین برد. بدین منظور نیاز است تا گرامر به نحوی بازنویسی شود که اولویت‌ها را لحاظ کند.

(آ) گرامر فوق را به گونه‌ای بازنویسی نمایید که عملگر + تقدم بیشتری نسبت به * و / داشته باشد. * و / هم تقدم بوده و تقدمشان از - بیشتر باشد. همچنین شرکت‌پذیری + و - از چپ به راست و شرکت‌پذیری * و / از راست به چپ باشد.

(ب) براساس گرامر قسمت (آ) مقدار عددی عبارت زیر را با تشکیل درخت تجزیه نحوی مشخص نمایید:

$$4 + 3 * 2 - 4/4$$

۶. گرامر زیر برای توصیف صوری عبارتهای ریاضی را در نظر بگیرید (غیرپایانه‌ها با حروف بزرگ شروع شده‌اند):

$$E \rightarrow E + T$$

$$| T - E$$

$$| T$$

$$T \rightarrow \text{no}$$

(آ) در صورت وجود درخت یا درخت‌های تجزیه نحوی را برای عبارت $2 - 3 + 4$ رسم و مقدار عبارت را در هر تجزیه حساب کنید.

(ب) در صورت وجود درخت یا درخت‌های تجزیه نحوی را برای عبارت $2 + 3 - 4$ رسم و مقدار عبارت را در هر تجزیه حساب کنید.

(ج) با توجه به قسمت‌های (آ) و (ب) شرکت‌پذیری از راست به چپ و از چپ به راست در چگونگی تفسیر عبارتهای ریاضی یک زبان را چگونه توجیه می‌کنید؟

بخش سوم: تمرین برنامه‌نویسی

پرسش ۷ پیاده‌سازی یک تحلیل‌گر نحوی را با استفاده از ابزار ANTLR مطرح می‌کند.

۷. ابتدا نرم‌افزار ANTLR را به همراه اسلایدهای معرفی آن، از وب‌سایت درس دریافت و مطالعه نمایید.

● دریافت ANTLR^۵

● اسلایدهای کار با ANTLR^۶

می‌خواهیم با استفاده از نرم‌افزار ANTLR یک تحلیل‌گر نحوی برای زبان MiniJava ایجاد کنیم. گرامر و سایر توضیحات این زبان از لینک زیر قابل دریافت است.

<http://www.cambridge.org/us/features/052182060X/grammar.html>

تحلیل‌گر نحوی شما بایستی برای هر رشته ورودی (یک برنامه ساده به زبان جاوا)، درخت تجزیه نحوی آن را رسم کرده و تعلق یا عدم تعلق رشته به زبان را مشخص نماید.

^۵<http://wwwantlr.org/>

^۶http://parsa.iust.ac.ir/download_center/courses_material/compilers/slides/IntroductionToAntlr_presentation_v1.pdf

تذکرات مهم:

۱. تمرین‌های کلاس حل تمرین با شماره‌های متوالی (hw01 ، hw02 و غیره) در وبسایت رسمی درس به نشانی <http://parsa.iust.ac.ir/courses/compilers/> منتشر می‌شوند و تنها راه اطلاع از تمرین‌ها وبسایت مذکور خواهد بود. مسئولیت عدم اطلاع از اخبار کلاس حل تمرین بر عهده دانشجو است.
۲. تمرین‌ها بایستی به صورت انفرادی حل شوند. در صورت مشاهده تقلب، نمره منفی به افراد متخلف تعلق می‌گیرد.
۳. پاسخ کل پرسش‌های هر تمرین را در قالب یک فایل فشرده (.zip) تحت عنوان compiler971_EN_SN به نشانی ایمیل compiler_iust@outlook.com ارسال نمایید که EN شماره تمرین (hw01 ، hw02 و غیره) بوده و SN شماره دانشجویی شما است. به عنوان مثال دانشجویی با شماره دانشجویی ۹۰۷۲۳۰۴۳، فایل پاسخ تمرین سری اول را تحت عنوان compiler971_hw01_90723043 ارسال خواهد کرد.
۴. عنوان ایمیل ارسالی دقیقاً عنوان فایل فشرده شده باشد، در غیر این صورت ایمیل شما دریافت نخواهد شد. در متن ایمیل نیز نام، نام‌خانوادگی و شماره دانشجویی خود را بنویسید.
۵. برای پرسش‌های حل‌کردنی (تشریحی) یک فایل PDF حاوی پاسخ همه پرسش‌ها ایجاد کنید. بهتر است پاسخ تمرین‌های تشریحی تایپ شده باشند؛ اما، می‌توانید پاسخ دست‌نویس خود را با استفاده از ابزارهایی مانند CamScanner تبدیل به فایل PDF کنید. در هر صورت پاسخ شما بایستی خوانا باشد.
۶. برای پرسش‌های برنامه‌نویسی کلیه فایل کدهای نوشته‌شده (فقط کد منبع) را به همراه تصاویری از خروجی اجرای برنامه، داخل یک فولدر مجزا قرار دهید و سپس ارسال نمایید. زبان برنامه‌نویسی دلخواه است، فقط کد خود را به نحو مقتضی توضیح‌گذاری^۷ نمایید.
۷. در صورتی که حجم فایل ارسالی شما زیاد بوده یا حاوی فایل اجرایی است، می‌توانید ابتدا آن را روی DropBox بارگذاری کرده و لینک آن را ایمیل نمایید. لطفاً توجه داشته باشید که لینک ارسالی شما باید تا پایان نیمسال فعال و ثابت بماند تا هر زمان توسط ما قابل بررسی و رؤیت باشد.
۸. منظور از مهلت ارسال پاسخ در سربرگ فایل تمرین ارسالی، حداکثر تاریخ جهت ارسال پاسخ شما است و ایمیل‌های ارسالی پس از این تاریخ بررسی نمی‌شوند.
۹. در تحویل حضوری پروژه پایانی از تمرین‌های طول ترم نیز سؤال پرسیده خواهد شد. همچنین لازم است دانشجویان محترم پاسخ تمرین‌های خود را همراه داشته باشند.
۱۰. برای ارتباط با دستیاران حل تمرین از ایمیل compiler_iust@outlook.com استفاده کنید. در صورت نیاز به مراجعه حضوری روزهای دوشنبه ساعت ۱۲ تا ۱۳ به آزمایشگاه تحقیقاتی مهندسی معکوس واقع در طبقه دوم دانشکده مراجعه نمایید.

⁷Comment