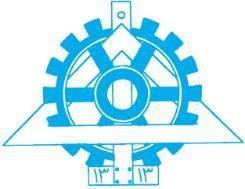
به نام خدا

فاز سوم پروژه کامپایلرها و زبان‌های برنامه‌نویسی

بهار 1400

مهلت تحویل : 16 خرداد

در این فاز پروژه، قرار است تحلیل معنایی زبان Jepeto را که در فاز قبلی آغاز کردید به اتمام برسانید. بدین منظور شما قوانین بررسی نوع را پیاده سازی خواهید کرد. برای تعیین قوانین از اطلاعات بیان شده در ادامه استفاده کنید.

**نوع**

همانطور که میدانید هر نوع زیرنوع خودش است (رابطه انعکاسی میان نوع ها وجود دارد). به

عنوان مثال int<:int است.

**Overloading**

در زبان Jepeto تنها عملگرهای is و not و تابع print به صورت overloaded تعریف شده اند. در عبارات e1 is e2 و e1 not e2 باید نوع e1 و e2 یکسان باشد. این دو عملگر روی تمام نوع ها بجز لیست تعریف شده‌اند. توجه داشته باشید در صورتی که یک عملوند آن‌ها از نوع اشاره‌گر تابع باشد، عملوند دیگر می‌تواند void نیز باشد.

در فراخوانی (print(e عبارت e میتواند از نوع int ، string،bool یا لیست باشد.

عملگرهای دیگر را طبق جدول زیر بررسی کنید:

|  |  |
| --- | --- |
| عملگرها | نوع عملوند(ها) |
| ضرب(\*)، تقسیم(/)، جمع(+)، تفریق(-)، منفی تک عملوندی(-)، بزرگتر(<)، کوچکتر(>) | int |
| عطف منطقی(and)، فصل منطقی(or)، نقیض منطقی(~) | bool |
| الحاق(append) | عملوند سمت چپ : list  عملوند سمت راست : هم نوع با نوع لیست |
| سایز (size.) | list |

مشخص کردن نوع ها

در این فاز قبل از اینکه بتوانید خطاهای نوع را بررسی کنید، لازم است که نوع بازگشتی توابع و آرگومان‌هایشان را به دست آورید. بدین منظور با شروع از main توابعی که فراخوانی شده اند را ویزیت کنید و با توجه به مقادیری که با آن فراخوانی شده‌اند، نوع آرگومان‌ها را مشخص کنید. هم چنین نوع عبارت استفاده شده در گزاره return، نوع بازگشتی تابع خواهد بود. پس از بررسی تمام توابع فراخوانی شده، به بررسی خطاها بپردازید. توجه کنید که نیازی به مشخص کردن نوع توابعی که فراخوانی نشده‌اند، نیست و خطاهای درون این توابع نیز مورد بررسی قرار نمی‌گیرند.

برای هر تابع به ترتیب تعریف شده در برنامه موارد زیر را چاپ کنید:

<function name>

<types of arguments list>

<return type>

مثال:

f

[IntType, BoolType]

IntType

لیست خطاها

در این فاز لازم است در صورتی که در بررسی های نام خطایی وجود نداشت، موارد زیر را بررسی کنید و در صورت خطا، پیام مناسب را به فرمتی که برای هر خطا گفته شده چاپ کنید و به بررسی کد تا انتهای فایل ادامه دهید. برای اینکه خطا در یک قسمت روی تمام قسمت های دیگر اثر نگذارد، فرض می‌کنیم که تمامی متغیرهایی که در نوع خود خطا دارند از یک نوع پیش فرض به نام NoType هستند. **NoType زیرنوع تمامی نوع‌های دیگر است. نوع عباراتی که در نوع خود خطا دارند نیز از همان نوع NoType است**.

فرمت کلی هر یک از خطاهایی که باید نمایش دهید به شکل زیر است:

Line <LineNumber>: <ErrorItemMessage>

مثال:

Line 21: List index is not an integer

که ErrorItemMessage برای برقرار نبودن هر یک از اصول زیر بدین صورت چاپ می‌شود :

1. **مجاز بودن نوع هنگام استفاده از عملگرها**

در عملگرهای دوتایی، در صورتی که یک عملوند از نوع NoType باشد، این خطا باید

برای عملوند دیگر بررسی شود.

ErrorItemMessage: Unsupported operand type for operator <OperatorName>

1. **استفاده از نوع bool در شرط ساختار تصمیم گیری**

نوع شرط در عبارت if باید از نوع bool باشد.

ErrorItemMessage: Condition must be bool

1. **مجاز بودن نوع ورودی تابع print**

ErrorItemMessage: Unsupported type for print

1. **عدم استفاده از مقدار بازگشتی تابعی که نوع بازگشتی آن void تعریف شده است**

ErrorItemMessage: Cannot use return value of a function having void return type

1. **بررسی نوع مقدار بازگشتی در هنگام استفاده از دستور return با نوع بازگشتی تابع**

نوع مقدار بازگشتی باید زیرنوع نوع بازگشتی تابع باشد.

ErrorItemMessage: Return value does not match with function return type

1. **عدم دسترسی به عناصر متغیری که از نوع لیست نباشد**

ErrorItemMessage: Access by index on a non-list

1. **integer بودن نوع اندیس در هنگام دسترسی به عناصر لیست**

ErrorItemMessage: List index is not an integer

1. **عدم استفاده از فیلد سایز بر متغیری که از نوع لیست نباشد**

ErrorItemMessage: Get size of a non-list

1. **عدم تعریف لیست با عناصر از نوع های مختلف**

ErrorItemMessage: Elements of the list have different types

1. **قابل فراخوانی بودن در هنگام فراخوانی (امتیازی)**

فراخوانی باید روی اشاره‌گر تابع و یا تابع ها صورت گیرد.

ErrorItemMessage: Calling a non-callable

1. **تطابق آرگومان های توابع با تعریف آن ها در هنگام فراخوانی (امتیازی)**

نوع آرگومانهای متناظر باید زیرنوع نوع تعریف شده برای آن در تعریف تابع باشد.

ErrorItemMessage: Args in function call do not match with definition

نکته: در صورت وجود بعضی خطاها در یک متغیر، خطاهای مرتبط نباید بررسی شوند، مانند:

* در صورت وجود خطای قسمت 3، خطاهای قسمتهای 5 و 7 برای آن فراخوانی بررسی نمی‌شوند.

همچنین در حالتی که پس از پیمایش کامل کد ورودی هیچ خطایی نداشته باشد، برنامه شما باید خروجی زیر را چاپ کند:

Compilation successful

**نکات مهم:**

* در این فاز شما باید کدهای داده شده را تکمیل کنید. خطاها و visitor ها به شما داده شده و باید از آنها استفاده کنید. کلیه فایل ها را به صورت یک فایل P3\_<studentID1>\_<studentID2>.zip آپلود کنید (کدهای جدول علائم، گرامر و visitor را هم در آن قرار دهید). توجه شود که تنها یک نفر از هر گروه باید پروژه را آپلود کند.
* در صورت کشف هرگونه تقلب، نمره 100- لحاظ می‌شود.
* دقت کنید که خروجی‌های شما به صورت خودکار تست می‌شوند؛ پس نحوه چاپ خروجی شما باید عیناً مطابق موارد ذکر شده در بالا باشد. تنها موارد خواسته شده را در فایل خروجی نمایش دهید و از قرار دادن خط‌های خالی و فاصله و ... نیز خودداری کنید.
* بهتر است سوالات خود را در فروم یا گروه درس مطرح نمایید تا دوستانتان نیز از آنها استفاده کنند؛ در غیر این صورت به مسئولان پروژه ایمیل بزنید:

[arash3908@gmail.com](mailto:arash3908@gmail.com) آرش رسولی

[nazaninyousefian@ut.ac.ir](mailto:nazaninyousefian@ut.ac.ir) نازنین یوسفیان