



## ۱. مقدمه

در این تمرین شما باید برخی از قسمت‌های مربوط به تولید کد را به کامپایلر زبان LogicPL اضافه کنید به طوری که کامپایلر بتواند فایل‌های class را ایجاد کند. برای تولید فایل‌های class می‌توانید از **Jasmin assembler** استفاده کنید. برای ساده‌تر شدن تمرین، فقط کافی است کد مربوط به حالتی که در ادامه توضیح داده شده را تولید کنید.

## ۲. ساختار حالتی که کامپایلر باید بتواند کد آن را تولید کند

در ابتدای برنامه، یک تابع به نام **calculate** تعریف می‌شود. در قسمت main برنامه فقط تابع calculate فراخوانی می‌شود. تابع **calculate** فاقد آرگومان ورودی است. در بدنه‌ی این تابع، یک متغیر به نام **result** (از نوع **int**) تعریف می‌شود، مقدار result با استفاده از عملگرهای محاسباتی (جمع، ضرب، تفریق، تقسیم و باقیمانده‌ی تقسیم) محاسبه می‌شود و در انتها مقدار این متغیر return می‌شود. بنابراین بدنه‌ی تابع calculate، فقط از گزاره‌های تعریف متغیر، assignment statement و return statement تشکیل شده است (گزاره‌های implication، تعریف predicate و loop در این تابع استفاده نشده است). فرض می‌شود که همه‌ی متغیرها و اعداد مورد استفاده در تابع calculate از نوع **int** باشند.

مثالی از برنامه‌هایی که این کامپایلر باید بتواند کد آنها را تولید کند به صورت زیر است:

```
function calculate(): int {  
    int p = 20;  
    int q = 4;  
    int result;  
    result = 3 + p / q;  
    return result;  
}  
main{  
    calculate();  
}
```

### ۳. نکات پایانی

- سوالات خود را در حد امکان در گروه درس مطرح نمایید.
- توجه کنید که در صورت احراز تقلب به هر نحو (دسترسی به کد دیگران، استفاده از کدهای ترم‌های قبل، استفاده از کدهای موجود بصورت آنلاین و ...)، مطابق قوانین درس با آن برخورد خواهد شد.
- کد خود را به صورت یک فایل zip با نام <sid2>-<sid1>-PLC-Phase4 آپلود کنید.
- آپلود فایل نهایی پروژه فقط توسط یکی از اعضای گروه صورت پذیرد.

موفق باشید.