

امتحان میان ترم اصول طراحی کامپایلر

بارم بندی امتحان (از 100 نمره):

- نوشتن صحیح گرامر : 20 نمره
- اعمال تغییرات لازم در کلاس CustomASTListener : 15 نمره
- اعمال تغییرات لازم در کلاس IMapper : 35 نمره
- ایجاد یک فایل ورودی حاوی حداقل 3 مثال پیچیده جهت ارزیابی تغییرات اعمال شده: 10 نمره
- اجرای صحیح کد پس از تبدیل به فایل اجرایی: 10 نمره
- تمیز و خوانا بودن کد: 10 نمره
- طراحی ورودی بسیار پیچیده و خلاقانه جهت ارزیابی تغییرات : 10 نمره اضافی (در صورت اجرای موفق برنامه)
- تهیه گزارش از تغییرات اعمال شده به صورت فایل PDF : 10 نمره اضافی (در صورت اجرای موفق برنامه)

سوال: با اعمال تغییرات لازم در پروژه IL Generator که سورس کد آن در دسترس شما قرار دارد، امکان استفاده از Switch case statement را به این پروژه بیافزایید. می توانید به دلخواه یکی از انواع Switch case statement هایی را که در ذیل مشاهده می نمایید را جهت پیاده سازی انتخاب کنید.

نوع 1: switch case statement زبان C: در این حالت case ها به ترتیب بررسی میشوند، اگر یک case مطابقت یابد تا زمانی که دستور break اجرا نشود تمامی case های بعدی نیز اجرا می شوند. مقدار یک متغیر از نوع int به switch داده می شود و case ها یک expression را مطابقت می دهند:

```
switch (a)
begin
    case 2+3:
        begin
            b:=5+ 8
            output:=a+9
        end
        break
    case 8*2:
        output := 9*5
    case 56:
        output:= 86
end
```

پس از اجرای جمله بالا تغییرات زیر باید حاصل شوند:

a=5 → b=13, output=14

a=16 → output=86

a=56 → output=86

نوع 2: switch case statement دارای مقدار بازگشتی: در این حالت case ها به ترتیب بررسی میشوند، اگر یک case مطابقت یابد دستور العمل های درون آن case اجرا شده و یک مقدار توسط کلمه کلیدی return برگشت داده می شود. مقدار یک متغیر از نوع int به switch داده می شود و case ها یک expression را مطابقت می دهند:

```
c := switch (a)
begin
    case 2+3:
        begin
            b:=5+ 8
            return b+9
        end
    case 8*2:
        return a+6
    case 56:
        return 8
end
```

پس از اجرای جمله بالا تغییرات زیر باید حاصل شوند:

a=5 → b=13, c=14

a=16 → c=22

a=56 → c=8

نوع 3: switch case statement مشابه زبان c (دارای break ضمنی): در این حالت case ها به ترتیب بررسی میشوند، اگر یک case مطابقت یابد فقط آن case اجرا می شود. مقدار یک متغیر از نوع int به switch داده می شود و case ها یک expression را مطابقت می دهند و اگر هیچ کدام از case ها مطابقت نیابند default case اجرا می شود:

```

switch (a)
begin
    case 2+3:
        begin
            b:=5+ 8
            output:=a+9
        end
    case 8*2:
        output := 9*5
    case 56:
        output:= 86
    case default:
        output:=78
end

```

پس از اجرای جمله بالا تغییرات زیر باید حاصل شوند:

a=5 → b=13, output=14

a=16 → output=45

a=56 → output=86

a not in [5, 16, 56] → output=78

موفق باشید