

Chapter 5 Mitchell

5.1

```
Integer procedure Q(I)
  Integer I;
  Begin
    Print (2+I);
  End;
```

چون در Algol 60 اگر یک تابع به عنوان ورودی به یک تابع دیگر داده شود، نوع ورودی های داده شده به تابع ورودی بررسی نمی شود ممکن است run-time error رخ دهد. در این مثال تابع Q به تنهایی از نظر type مشکلی ندارد ولی در تابع P ورودی bool به این تابع داده می شود و در این تابع با یک مقدار integer جمع می شود و run-time error رخ می دهد.

5.3

a)

```
fun f(p) = if #2(p)=0 then #1(p) else if #1(p)=0 then #2(p) else
#1(p)+#2(p);
```

b) yes

c)

باید اسامی متغیر های یکسان را تغییر دهیم و در شرط ها برابری آن ها را بررسی کنیم.

```
Fun eq(x,y) = if x=y then true else false
```

d)

5.4

a)

```
Fun maptree(f,leaf(y))= f(y)
  | maptree(f,node(y,z)) = node(maptree(f,y),maptree(f,z))
```

b)

تابع f می تواند type را تغییر دهد.

$(a \rightarrow b) \rightarrow a \text{ tree} \rightarrow b \text{ tree}$

5.6

a)

Fun C1(x,y)=z

Fun C2(x)=C3

Fun C3(y)=z

Fun Curry(C1)=C2

Fun U1(x,y)=z

Fun U2(x)=U3

Fun U3(y)=z

Fun UnCurry(U2)=U1

b)

چون ورودی تابع uncurry و خروجی تابع curry دارای type یکسان هستند و هم چنین خروجی تابع uncurry و ورودی تابع curry یکسان هستند وقتی این دو تابع رو به ترتیب روی f و g اپلای کنیم نتیجه با f و g برابر است.

5.7

a)

چون بررسی نمی شود که ایا قسمت integer حتما مقدار دهی اولیه شده است یا نه (ممکن است در قسمت if ، else اجرا شود)

b)

در ML به دلیل این که conservative است مشکل قبلی پیش نمی آید و در زمان کامپایل به کاربر warning می دهد که برنامه ای که نوشته است safe نیست.