



تاریخ: ۵ تیر ۱۳۹۷

مدت امتحان: ۱۲۰

مدرس: مجتهدی

## آزمون پایان ترم کامپایلر

۱. (۳۰ نمره) (آ) (۱۵ نمره) منظور از زبان statically typed و dynamically typed چیست؟ شرح دهید که هر یک چه مزیتی نسبت به دیگری دارد. همچنین برای هر کدام مثالی بیان کنید.  
(ب) (۱۵ نمره) تمایز بین تایپ پایا (static) و تایپ پویا (dynamic) چیست؟
۲. (۳۰ نمره) (آ) (۱۵ نمره) صحت و تمامیت برای یک کامپایلر به چه معنا است؟ هر کدام را به تفکیک بیان کنید.  
(ب) (۱۵ نمره) آیا کامپایلر موجود برای COOL صحیح است؟ آیا کامل است؟ برای پاسخ خود استدلال بیاورید.
۳. (۲۰ نمره) توضیح دهید که SELF\_TYPE چیست و با یک مثال از زبان COOL شرح دهید که چه ضرورتی برای SELF\_TYPE وجود دارد؟
۴. (۲۰ نمره) کد MIPS برای دستور زیر را بنویسید:

```
def f(n) = if n=0 then 0 else n+ f(n-1)
```

۵. (۱۰ نمره) بهینه‌سازی سرتاسری و موضعی چیست؟ تعریف هر کدام را بیان کنید.
6. (20 points) Suppose we extend Cool with exceptions by adding two new constructs to the Cool language.

```
expr ::= try expr catch ID => expr  
      | throw expr  
      | ...
```

Here **try**, **catch** and **throw** are three new terminals. “**throw** *expr*” returns *expr* to the closest dynamically enclosing catch block. Note that since **throw** expression returns control to a different location, we do not really care about the context in which throw is used. For example, (**throw** *false*) + 2 is a valid Cool expression (However, note that (**throw** *false*) + (2 + *true*) is not a valid Cool expression). Following is an example that uses the try-catch and throw constructs.

```
try  
  if some_test1 then throw 34  
  else if some_test2 then throw ``undefined error''  
  else do_something fi fi
```

```
catch x =>
  case x of
    x:Int => do_something1
    x:String => do_something2
  esac
```

The above program fragment executes “do\_something1” (with  $x$  bound to the value 34) if “some\_test1” evaluates to *true*. It executes “do\_something2” (with  $x$  bound to the value “undefined error”) if “some\_test1” evaluates to *false* but “some\_test2” evaluates to *true*. It executes “do\_something” if both “some\_test1” and “some\_test2” evaluate to *false*.

Give a set of new sound typing rules that we can add to the Cool specification to accommodate these two new constructs.

موفق باشید.