

تمرین‌های برنامه‌نویسی درس اصول طراحی کامپایلر

کامپایلر تسلنگ: گام اول

در گام اول از تمرین عملی درس طراحی کامپایلر، باید یک تحلیلگر لغوی بنویسید. در این گام باید برنامه‌ای بنویسید که با خواندن یک فایل در زبان تسلنگ از ورودی استاندارد، واژه‌های (Tokens) آن را چاپ کند. برای مثال، کد تسلنگ زیر را در نظر بگیرید:

```
function example:
    param list A;
    returns num;
{
    return (listlen(A) + 1);
}
```

برای مثال، اگر این قطعه کد به عنوان ورودی داده شود، برنامه‌ی شما باید هر واژه را در یک خط خروجی چاپ نماید:

```
function
example
:
param
list
A
;
returns
num
;
{
return
(
listlen
(
A
)
+
1
)
;
}
```

دقت کنید که واژه‌ها ممکن است با فاصله جدا نشده باشند. این تمرین تا تاریخ ۹۹/۱/۳۰ قابل انجام است.

کامپایلر تسلنگ: گام دوم

در گام دوم از تمرین عملی درس طراحی کامپایلر، تحلیل نحوی را انجام می‌دهید. در این گام برنامه‌ای می‌نویسید که با خواندن یک فایل تسلنگ از ورودی استاندارد و تحلیل نحوی آن، پیغام‌هایی را چاپ می‌کند. دقت کنید که تجزیه‌ی برنامه‌ی ورودی الزامی است و منطق برنامه باید به کمک عملیات مفهومی نوشته شود. پیغام‌ها متغیرهایی را گزارش می‌کند که در هنگام استفاده مقدار دهی نشده‌اند. برای مثال، فایل زیر را در نظر بگیرید:

```
1  function sum3:
2      param num a;
3      param num b;
4      param num c;
5      returns num;
6  {
7      local num sum;
8      c = a + b + c;
9      return sum;
10 }
11
12 function main:
13     returns num;
14 {
15     local num a;
16     local num b;
17     local num c;
18     local num x;
19     a = 5
20     c = numread();
21     a = sum3(a, b, c);
22     numprint(x);
23     return 0;
24 }
```

برنامه‌ی شما پس از خواندن این فایل باید خط‌های زیر را چاپ نماید؛ چاپ کردن شماره‌ی خط الزامی نیست.

```
9: <sum> is used uninitialized!
21: <b> is used uninitialized!
22: <x> is used uninitialized!
```

این تمرین تا تاریخ ۹۹/۳/۶ قابل انجام است.

کامپایلر تسلنگ: گام سوم

در گام سوم از تمرین عملی درس طراحی کامپایلر، برنامه‌ای می‌نویسید که با خواندن یک فایل تسلنگ از ورودی استاندارد، کد میانی آن را تولید می‌کند. برای نمونه، کد زیر را که در زبان تسلنگ است در نظر بگیرید.

```
function sum3:
    param num a;
    param num b;
    param num c;
    returns num;
{
    local num sum;
    sum = a + b + c;
    return sum;
}

function main:
    returns num;
{
    local num a;
    local num b;
    local num c;
    a = numread();
    b = numread();
    c = numread();
    numprint(sum3(a, b, c));
    return 0;
}
```

برنامه‌ی شما پس از خواندن این فایل باید کد میانی تسلنگ را تولید کند. یک خروجی نمونه برای این دو تابع در ادامه نشان داده می‌شود.

```
proc sum3
    add r0, r0, r1
    add r0, r0, r2
    ret

proc main
    call iget, r3
    call iget, r1
    call iget, r2
    call sum3, r3, r1, r2
    call iput, r3
    mov r0, 0
    ret
```

این تمرین تا تاریخ ۹۹/۳/۳۰ قابل انجام است.