



کامپایلر و زبانهای برنامه نویسی (کارشناسی)

مدرس: ذاکری

دانشکده فنی فومن

آزمون میان ترم (درس های یک تا چهار)

تاریخ آزمون: آذر

نیمسال تحصیلی: اول (پاییز ۱۴۰۲)

سال تحصیلی ۱۴۰۲-۰۳

مدت پاسخ گویی: ۹۰ دقیقه

توجه:

- دفترچه سؤالات آزمون شامل ۳ صفحه و ۶ پرسش است.
 - پاسخ ها به ترتیب پرسش ها نوشته شده و همه مراحل راه حل طی شده نیز نوشته شوند.
- نام و نام خانوادگی: شماره دانشجویی:

پرسش	۱	۲	۳	۴	۵	۶	جمع نمرات
بارم	۲۰	۲۰	۲۰	۱۵	۱۵	۱۰	۱۰۰
نمره							

۱. (۲۰ امتیاز) درست یا نادرست بودن هریک از گزاره های زیر را با ذکر دلیل مشخص نمایید:
 - (آ) در بازنویسی کامپایلر یک زبان برنامه نویسی برای یک ماشین جدید، واحدهای تحلیل گر نحوی و مفهومی نیازی به تغییر ندارند.
 - (ب) تحلیل مفهومی کندترین مرحله از مراحل یک کامپایلر از نظر زمان اجرا است.
 - (ج) زبان برنامه نویسی Python، مستقل از متن است.
 - (د) برای زبان $L = \{w \in \{0, 1\}^* : n_0(w) = n_1(w)\}$ گرامر غیر مبهم وجود دارد.
 - (ه) ساختار پرانتزها در عبارتهای ریاضی، با زبان مستقل از متن $L = \{w \in \{(\,,)\}^* : n_0(w) = n_1(w)\}$ قابل توصیف است.
۲. (۲۰ امتیاز) گرامر عبارتهای ریاضی، با فرم کلی زیر را در نظر گرفته و به پرسش های داده شده پاسخ دهید.

$$E \rightarrow E \text{ Op } E \mid (E) \mid id \mid num$$

$$\text{Op} \rightarrow + \mid - \mid * \mid / \mid \wedge$$

- (آ) یک گرامر معادل با گرامر فوق، در فرم LL(1)، ارائه دهید که در آن عملگر + تقدم بیشتری نسبت به * و / داشته باشد. * و / هم تقدم بوده و تقدمشان از - بیشتر باشد. همچنین، شرکت پذیری + و - از چپ به راست و شرکت پذیری * و / از راست به چپ باشد.



(ب) حداکثر طول جمله‌گونه (sentential form) در تحلیل نحوی عبارت " $4 + 3 * 2 - 4/4$ " با تجزیه‌کننده پیشگو چند است؟

(ج) مقدار عبارت " $4 + 3 * 2 - 4/4$ " را با رسم درخت تجزیه نحوی، از روی گرامر LL(1)، ارزیابی کنید.

۳. (۲۰ امتیاز) جدول تجزیه LL(1)، با قابلیت بهبود از خطا (error recovery) به روش panic-mode را برای گرامر زیر تشکیل داده و تعداد خانه‌های خطا را مشخص نمایید.

$$\begin{aligned} S' &\rightarrow S\$ \\ S &\rightarrow AB \\ A &\rightarrow aA \\ A &\rightarrow a_1A \\ A &\rightarrow D \\ B &\rightarrow bB \\ B &\rightarrow b_1B \\ B &\rightarrow \lambda \\ D &\rightarrow dD \\ D &\rightarrow \lambda \end{aligned}$$

۴. (۱۵ امتیاز) خطاهای موجود در برنامه ۱، در زبان C++ استاندارد، و نوع هر خطا را در قالب یک جدول به ترتیب زمان تشخیص آنها، مشخص کنید.

```
1 include <iostream>
2 void setArray() {
3     const float f = 10.5; const int i = 20
4     int j = 20;
5     int A[j], B[i];
6     B[30] = 15;
7     B[j] = i;
8     B[0] = 1.5;
9     B[1] = i;
10    A[x] = 15;
11    A[19.5] = 15;
12    B[f] = -1;
13    int re@ = 2**3 + 1;
14    j = f + i;
15    return 0;
16 } // end of the function
```

برنامه ۱: تابع setArray نوشته شده با C++.

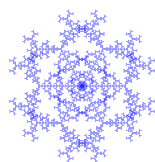
۵. (۱۵ امتیاز) یک گذارنما (transition diagram) برای تشخیص توضیحات چند خطی (multi-line comments) در زبان‌های خانواده C با کمترین تعداد حالت، طراحی کنید. چند گذر و چند حالت با نماد * در این نمودار وجود دارد؟ (توجه: حالت‌هایی که نیاز به retract دارند، در صورت وجود، با * مشخص می‌شوند.)



۶. (۱۰ امتیاز) توصیف بخشی از Scanner زبان فرضی JavaScript در FLex به شرح ذیل (برنامه ۲) داده شده است. خروجی این Scanner برای رشته "===={Hello}===={World}=====" را مشخص کنید.

```
1 %{ #include <stdio.h> %}  
2 /* Scanner for a subset of the JavaScript language */  
3 EQ      [=]  
4 %%  
5 ==      {printf("Hello %s\n", yytext);}  
6 {EQ}{3} {printf("World %s\n", yytext);}  
7 "{ "[^}\n]*"}"  
8 [ \t\n]+  
9 .      printf("Not %s\n", yytext);  
10 %%  
11 int main( int argc, char **argv ){  
12     ++argv, --argc; /* skip over program name */  
13     if ( argc > 0 )  
14         yyin = fopen( argv[0], "r" );  
15     else  
16         yyin = stdin;  
17     yylex();  
18     return 0;  
19 }
```

برنامه ۲: توصیف Scanner زبان JavaScript.



موفق باشید.