

۳۰ - گزاره های زیر را در نظر بگیرید:

الف - هر زبان مستقل از متن را می توان به کمک گرامری تعریف کرد که قابلیت تجزیه رشته های ورودی در زمان خطی و به طور پایین به بالا را فراهم کند.

ب - حذف قوانین بازگشتی چپ هر گونه ابهام را برای تجزیه گرامرهای بالا به پایین از بین می برد.

ج - برای حذف ابهامات در یک پارسر بالا به پایین لازم و کافی است که پس از حذف قوانین بازگشتی چپ در گرامر حتماً عمل فاکتورگیری چپ را نیز انجام داد.

(۲) الف و ب درست هستند.

(۱) الف و ج درست هستند.

(۴) فقط ج صحیح است.

(۳) الف و ب و ج غلط هستند.

۳۱ - گرامر عبارات ریاضی G به شرح زیر و یکی از حالت های پارسر $SLR(1)$ مطابق شکل مفروض است که در آن ϵ رشته ای به طول صفر است. اگر پارسر در حالت داده شده باشد و واژه بعدی برنامه ورودی ("باشد، حرکت بعدی پارسر کدام است؟

$\begin{aligned} E &\rightarrow T.E' \\ E' &\rightarrow \epsilon \\ E' &\rightarrow . + TE' \end{aligned}$
--

$$G : E \rightarrow TE'$$

$$E' \rightarrow +TE'$$

$$E' \rightarrow \epsilon$$

$$T \rightarrow FT'$$

$$T' \rightarrow *FT'$$

$$T' \rightarrow \epsilon$$

$$F \rightarrow id$$

$$F \rightarrow (E)$$

(۲) shift به حالت بعدی با ϵ

(۱) Reduce با Push کردن یک حالت در Stack

(۴) اعلام خطا

(۳) Reduce با POP کردن stack

۳۲ - در زبان هایی که اشاره گر (Pointer) ندارند در چه حالتی وقوع پدیده همنامی یا نام مستعار (Aliasing) امکان ندارد؟

(۱) اگر تنها تکنیک انتقال پارامتر در زبان مورد نظر $by\ value$ باشد.

(۲) اگر مجموعه متغیرهای سراسری (global) برنامه تهی باشد.

(۳) اگر زبان دارای تکنیک انتقال $by\ reference$ نباشد.

(۴) هیچکدام از سه گزینه فوق صحیح نیستند.

۳۳- زبان های زیر را از نظر تفسیری یا کامپایلری بودن دسته بندی کنید. دسته بندی درست را انتخاب کنید.

Ada - VI , Small talk - V - LISP - IV , C++ - III , Java - II , Fortran - I

(۱) VI , V , IV (۲) V , IV , II (۳) III , II , I (۴) هیچکدام

۳۴- در کدام یک از موارد زیر اگر تکنیک انتقال پارامتر by name یا by reference باشد نتیجه اجرای برنامه می تواند متفاوت باشد؟

(۱) پارامتر مربوطه متغیر ساده (مثلاً A) است.

(۲) پارامتر مربوطه عضو نامعینی از یک آرایه (مثلاً B[I]) است.

(۳) پارامتر مربوطه عضو معینی از یک آرایه است. (مثلاً B[5])

(۴) هر سه مورد

۳۵- برای ۳ دستور case به شرح زیر، پیاده سازی معروف به جدول پرش ها (Jump table) مفروض است؟

I. case I of \ : st_۱; ۲ : st_۲; ۳ : st_۳ end case

II. case I of ۱۰۰ : st_۱; ۲۰۰ : st_۲; ۳۰۰ : st_۳ end case

III. case I of \ : st_۱; ۲ : st_۲; ۳۰۰ : st_۳ end case

(۱) حجم کد I از II و II از III کمتر است.

(۲) سرعت اجرا و حجم کد هر سه دستور برابر است.

(۳) سرعت اجرای دستور I از III بیشتر و سرعت اجرای III از II بیشتر است.

(۴) هیچکدام

۳۶- در زبانی که قانون حوزه شناسایی ایستا حاکم است و تعریف تو در توی برنامه های فرعی (nested definition) ممکن است،

برنامه ای نوشته ایم که از یک برنامه اصلی حاوی تعریف دو برنامه فرعی غیر تو در تو تشکیل شده است. یکی از برنامه های فرعی

حای برنامه فرعی دیگری است که آن هم حاوی برنامه فرعی دیگری است. در یک لحظه از زمان اجرای داخلی ترین برنامه

فرعی، پشته (stack) رکوردهای فعالیت برنامه های فرعی حاوی ۱۰ رکورد (از جمله رکورد برنامه اصلی) است. در این لحظه

محتوای چند رکورد فعالیت قابل دسترسی توسط برنامه در حال اجرا است؟

(۱) ۳ (۲) ۴ (۳) ۱۰ (۴) هیچکدام

۳۷- در قطعه برنامه زیر که به زبان برنامه سازی ML نوشته شده است مقدار عبارت $b * f(a, a)$ در دو حالتی که زبان از قواعد حوزه ایستا (static scoping) و حوزه پویا (dynamic scoping) استفاده می کند، کدام است؟

`let a = 5, b = 10, c = 7 in`

(۱) حوزه ایستا: ۶۰ و حوزه پویا: ۳۴

`let val fun f(x, y) = a * (x + y) + b`

(۲) حوزه ایستا: ۵۰۰ و حوزه پویا: ۳۴۰

`let val a = 4, b = 2 in`

(۳) حوزه ایستا: ۱۰۰ و حوزه پویا: ۶۸

`b * f(a, a)`

(۴) هیچکدام

`end`

`end`

`end`
