Procesadores de Lenguaje. Junio 2009
Grupos A y C

Ejercicio 1 [1,5 puntos]
Encuentra los conjuntos **primeros** y **siguientes** para los símbolos no terminales de la siguiente gramática, así como la **tabla de análisis** para el analizador descendente predictivo no recursivo asociado. ¿Es una gramática LL(1)?

```
A --> B C a
B --> b | λ
C --> C D E | c | λ
D --> D d | d
E --> e | λ
```

Ejercicio 2 [1,5 puntos]
Considera la gramática

```
S --> a S b
S --> λ
```

Se pide:

a) **[0,5 puntos]** ¿Qué es un **prefijo viable**? ¿Qué es un **asidero**? ¿Cuál es la **propiedad básica** que cumplen los **prefijos viables** de una **gramática incontextual**? ¿Es "aaaSb" un prefijo viable de esta gramática? ¿y "aaaSbbb"? Justifica tus respuestas.

b) **[0,5 puntos]** Describe, **mediante una gramática**, el lenguaje de los prefijos viables de la siguiente gramática incontextual:

```
S --> a S b
S --> λ
```

c) **[0,5 puntos]** Ilustra los movimientos que realiza un analizador por **desplazamiento – reducción** en el reconocimiento de la sentencia "aabb".

Ejercicio 3 [2 puntos]

a) **[0,5 puntos]** ¿Qué es una gramática **s-atribuida**? Describe las transformaciones de eliminación de **recursión a izquierda directa** y de **factorización** en gramáticas **s-atribuidas**, especificando tanto la transformación de las producciones como la **transformación de las ecuaciones semánticas**.

b) **[1,5 puntos]** Tomando como referencia las transformaciones a las que se ha hecho alusión en el apartado a), factoriza, y también elimina la recursión a izquierdas directa, de la siguiente gramática de atributos, explicando las dificultades encontradas para la aplicación directa de las transformaciones:

```
A --> A a     A0.n = combina(A1.n,a.lex)
A --> A b     A0.n = junta(A1.n,b.lex)
A --> B c     A.n  = mezcla(B.n,c.lex)
A --> B d     A.n  = teje(B.n,d.lex)
B --> e       B.n  = inicia(e.lex)
```

(* CONTINUA EN EL REVERSO DE LA HOJA *)

Procesadores de Lenguaje. Junio 2009
Grupos A y C

Ejercicio 4 [5 puntos]
Considera el siguiente programa:

```
programa examen;

 tipo tPersona  = record
                    nombre: array [20] of char;
                    edad: int
                  end;

 tipo tListaPersonas =  ^tCeldaListaPersonas;
 tipo tCeldaListaPersonas  =  record
                                persona: tPersona;
                                sig: tListaPersonas
                              end;
 i: int;

 proc fuego(j: char; d: int);
 var
   k: int;
 begin
   read(k);
   writeln(ord(j)+d*k)
 end; (* de fuego *)


 proc tierra(j: char; peso: int);

       proc mar(k, m: int);
        var
          lista: tListaPersonas;

       begin   (*1*)
         new(lista);
         new(lista^.sig);
         lista^.sig^.persona.nombre[k+m] := j (*2*)
       end; (* de mar *)

 begin
   mar(i, i*peso) (*3*)
 end; (* de tierra *)

begin
  i := 1;
  fuego('J', 3);
  tierra('J', 2)
end.  (* de examen *)
```

Suponiendo que se cuenta con un traductor descendente que no realiza ningún tipo de optimización, se pide:

a) **[1 punto]** Describe el contenido de la tabla de símbolos cuando el traductor alcanza el punto (*1*)
b) **[1 punto]** Describe la arquitectura de una máquina-P dotada de manejo de memoria dinámica y que utiliza el mecanismo de **displays** para ejecutar subprogramas. Describe con el mayor grado de detalle posible el contenido de la memoria de la máquina-p inmediatamente después de ejecutar la instrucción (*2*)
c) **[1,5 puntos]** Representa mediante árboles las estructuras sintácticas de las sentencias (*2*) y (*3*), y marca sobre dichos árboles el recorrido que realiza el traductor.
d) **[1,5 puntos]** Escribe y comenta el código-p que resulta de la traducción de las sentencias (*2*) y (*3*). Indica claramente el propósito de los argumentos de cada instrucción en el código-p generado (¿es el argumento una dirección? ¿Es un nivel? ¿Es el contenido de una posición de memoria?, etc.). Sobre el recorrido de los árboles del apartado c) indica, así mismo, los puntos en los que se generan las distintas instrucciones.

(* CONTINUA EN EL REVERSO DE LA HOJA *)