
Lenguajes de Programación y Procesadores de Lenguajes

(2º parcial)

15 de enero de 2021

1. (2 ptos.) Dado el siguiente fragmento de un programa en C.

```
-----  
#include<stdio.h>  
int a ;  
  
int f1(int p1, float p2, int p3) {  
    float b ;  
  
    b = p2 * p1 +p3 ;      // <--- PUNTO DE CONTROL  
  
    return b ; }  
  
int main () {  
    int c = 2 ;  
  
    printf( "%d\n", f1(c, c*2, c*3)); }  
-----
```

- a) (0,75 ptos.) Mostrad el contenido completo de la TDS en el PUNTO DE CONTROL.
- b) (0,5 ptos.) Indicad el desplazamiento relativo de los parámetros p1 y p2 que aparecen en la instrucción del PUNTO DE CONTROL.
- c) (0,75 ptos.) Mostrad el contenido de los frames (RA) que se encuentran en la pila de ejecución cada vez que se pasa por el PUNTO DE CONTROL.

2. (1,5 ptos.) Contestad brevemente a las siguientes cuestiones:

- a) (1 pto.) Construid el grafo de interferencias para el siguiente bloque de instrucciones, sabiendo que las variables activas a la entrada son $\{y, d\}$ y a la salida son $\{x, y\}$

```
c = y * d  
x = y - 2  
b = x + y  
x = 5 * b  
y = c + x
```

- b) (0,5 ptos.) Indicad cual de las siguientes afirmaciones es FALSA (error = -0.33)
 - 1) El calculo previo de constantes es una transformación algebraica que mejora el código
 - 2) Durante la carga de un registro de activación la dirección de retorno la apila el bloque llamador.
 - 3) Para poder asignar memoria estática a un objeto es necesario conocer su tamaño en tiempo de compilación.
 - 4) El k-coloreado de un grafo de interferencias se emplea en el proceso de selección de instrucciones mediante revestimiento o análisis sintáctico

3. (3,5 ptos.) Dada la siguiente gramática,

$$\begin{aligned} I &\rightarrow \text{repeat } E : I \\ I &\rightarrow I ; I \\ I &\rightarrow \text{break} \end{aligned}$$

La instrucción **repeat** indica que la instrucción que le sigue, E, debe ejecutarse un total de E veces, donde E debe ser una expresión de tipo entero. La instrucción **break**, si aparece, supone la finalización del bucle. Diseñad un ETDS que realice las comprobaciones semánticas y genere código intermedio asociado.

4. (1 pto.) Dado el siguiente fragmento de código intermedio de un bloque básico, construid su GDA y aplicad las optimizaciones locales, a partir de este GDA. Considerad activas a la salida del bloque las variables: **A**, **x**.

(100) $t_1 = 0$	(105) $t_5 = x * 4$	(110) $t_9 = t_8 * 4$
(101) $k = t_1$	(106) $t_6 = A[t_5]$	(111) $A[t_9] = t_7$
(102) $t_2 = x + k$	(107) $t_7 = t_4 + t_6$	(112) $t_{10} = k * 4$
(103) $t_3 = t_2 * 4$	(108) $k = x$	(113) $t_{11} = A[t_{10}]$
(104) $t_4 = A[t_3]$	(109) $t_8 = k * 1$	(114) $x = t_{11}$

5. Dado el siguiente fragmento de código intermedio,

(100) $B = 0$	(104) $t_1 = i * 2$	(108) $t_3 = t_2 - B$	(112) $t_4 = t_2 - B$
(101) $N = 100$	(105) $M = A * 5$	(109) $z = X[t_3]$	(113) $z = Y[t_4]$
(102) $A = 2$	(106) $t_2 = t_1 + 4$	(110) goto 114	(114) $i = i + 1$
(103) $i = 0$	(107) if $t_2 > M$ goto 111	(111) $B = 10$	(115) if $i \leq N$ goto 104

- (0,5 ptos.) Determinad los bloques básicos que forman el bucle. Extraed el código invariante e indicad las variables de inducción y sus ternas asociadas.
- (0,75 ptos.) Aplicad el algoritmo de reducción de intensidad
- (0,75 ptos.) Aplicad el algoritmo de eliminación de variables de inducción.

1.

a)

TDS

TdArgumento

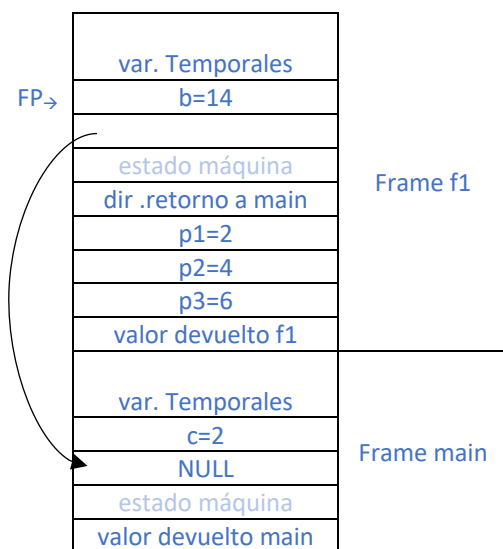
Lexema	Categoría	Tipo	Nivel	desp
a	variable	tentero	0	0
p1	parámetro	tentero	1	-10
p2	parámetro	treal	1	-14
p3	parámetro	tentero	1	-16
f1	función	tentero	0	--
b	variable	treal	1	0

→

tentero	treal	tentero
---------	-------	---------

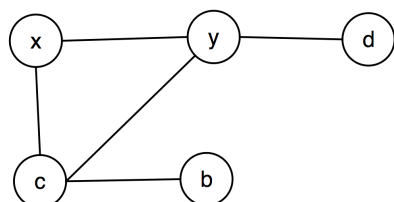
b) $\text{desp}_{p1}: -10$ $\text{desp}_{p2}: -14$

c)



2.a.-

	Def	Usa	Ent[]	Sal[]
<i>c:= y*d</i>	<i>c</i>	<i>y,d</i>	<i>y,d</i>	<i>c,y</i>
<i>x:= y-2</i>	<i>x</i>	<i>y</i>	<i>c,y</i>	<i>c,x,y</i>
<i>b:= x+y</i>	<i>b</i>	<i>x,y</i>	<i>c,x,y</i>	<i>c,b</i>
<i>x := 5*b</i>	<i>x</i>	<i>b</i>	<i>c,b</i>	<i>c,x</i>
<i>y:=c+x</i>	<i>y</i>	<i>c,x</i>	<i>c,x</i>	<i>x,y</i>

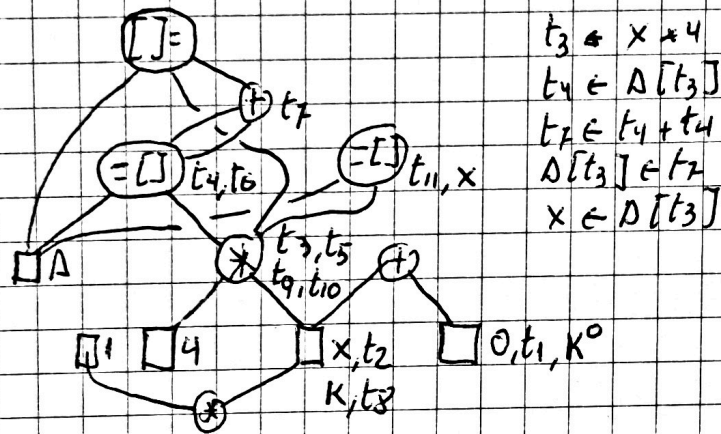


2.b.-

FALSA: “El *k*-coloreado de un grafo de interferencias se emplea en el proceso de selección de instrucciones mediante revestimiento o análisis sintáctico”

$I \rightarrow \text{repeat } E: \text{ Si } (E.t \neq \text{tentero}) \text{ Men Error}()$
 $I.\text{ini} \leftarrow \Omega; I.\text{pin} \leftarrow \text{creaPans}(\Omega); \text{emile}(\text{if } E.d = 0 \text{ goto } \square);$
 $I \text{ Emile}(E.d \leftarrow E.d - 1); \text{Emile}(\text{goto } I.\text{ini}); \text{completePans}(\text{fusionPans}(I.\text{pin}, I.b), \Omega); I.b \leftarrow \text{nil};$

$I \rightarrow I; I$
 $I \rightarrow \text{break}$

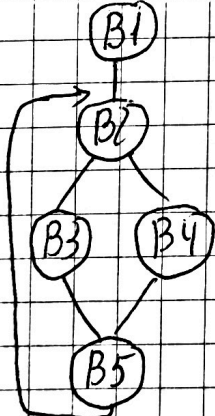


$t_3 \leftarrow x * 4$
 $t_4 \leftarrow A[t_3]$
 $t_7 \leftarrow t_4 + t_4$
 $A[t_3] \leftarrow t_7$
 $x \leftarrow A[t_3]$

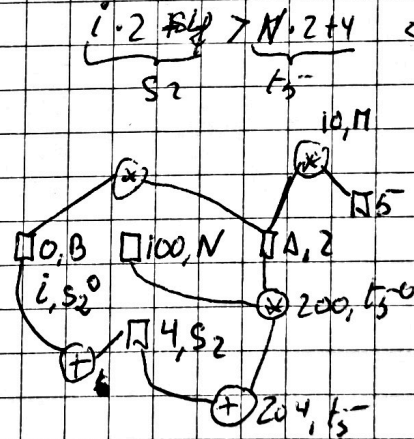
$B1$
 $B = 0$
 $N = 100$
 $A = 2$
 $i = 0$
 $M = A * 5$
 $S_1 = i * 2 \leftarrow 1$
 $S_2 = i * 2$
 $S_2 = S_2 + 4$
 $t_5 = N * 2$
 $t_5 = t_5 + 4$

$B2$
 $t_1 \leftarrow S_1 \leftarrow 1$
 $t_2 \leftarrow S_2 \leftarrow 3$
 $\text{if } t_2 > M \text{ goto } B4 \leftarrow$
 $B3$
 $t_3 = t_2 - B \leftarrow 3$
 $z = X[t_3]$
 $\text{goto } B5$
 $B = 10$
 $B4$
 $t_4 = t_2 - B \leftarrow 3$
 $z = Y[t_4]$
 $i = i + 1 \leftarrow 2$
 $B5$
 $S_1 = S_1 + 2 \leftarrow 1$
 $S_2 = S_2 + 2 \leftarrow 11$
 $\text{if } i \leq N \text{ goto } B2 \leftarrow 2$

$B1$
 $B = 0$
 $N = 100$
 $A = 2$
 $i = 0$
 $B2$
 $t_1 = i * 2$
 $M = A * 5$
 $t_2 = t_1 + 4$
 $\text{if } t_2 > M \text{ goto } B4$
 $B3$
 $t_3 = t_2 - B$
 $z = X[t_3]$
 $\text{goto } B5$
 $B = 10$
 $B4$
 $t_4 = t_2 - B$
 $z = Y[t_4]$
 $B5$
 $i = i + 1$
 $\text{if } i \leq N \text{ goto } B2$



$i'(i, 1, 0)$
 $t_1(i, 2, 0) S_1$
 $t_2(i, 2, 4) S_2$



$B2$
 $\text{if } S_2 > M \text{ goto } B4$
 $B3$
 $t_3 = S_2 - B$
 $z = X[t_3]$
 $\text{goto } B5$
 $B4$
 $B = 10$
 $t_4 = S_2 - B$
 $z = Y[t_4]$
 $B5$
 $S_2 = S_2 + 2$
 $\text{if } S_2 > t_5 \text{ goto } B2$
 $B1$
 $B = 0$
 $M = 10$
 $S_2 = 4$
 $t_5 = 204$