

Fundamentos de la programación y la informática

Examen de teoría. 21 de Diciembre de 2018

Grado en ingeniería aeroespacial en aeronavegación

Grado en ingeniería aeroespacial en vehículos aeroespaciales

Universidad Rey Juan Carlos

Preparativos

- Ejecuta el script `prepara_fpi`
- Esto creará en tu cuenta el directorio `~/fpi.diciembre.18` y el fichero `~/fpi.diciembre.18/teoria.TULOGIN.txt`, donde TULOGIN es tu nombre de usuario en el laboratorio. Escribe tus respuestas en este fichero.

Ejercicio 1

Sean a, b, c, d, e variables booleanas. Sean x e y variables reales.

Sean las expresiones

```
e1 := (a or b) and not ( (a and c) and (d or e) );
```

```
e2 := (x > 15) and not (y <= 23 );
```

1. A partir de $e1$, escribe una expresión lógica equivalente, más legible, de nombre $s1$.
2. A partir de $e2$, escribe una expresión lógica equivalente, más legible, de nombre $s2$.
3. Escribe una expresión $n1$ que sea la negación de $s1$.
4. Escribe una expresión $n2$ que sea la negación de $s2$.

Solución

```
s1 := (a or b) and (((not a) or (not c)) or ((not d) and (not e)));
```

```
s2 := (x > 15) and (y > 23);
```

```
n1 := ((not a) and (not b)) or ((a and c) and (d or e));
```

```
n2 := (x <= 15) or (y <= 23);
```

Ejercicio 2

Como sabes, en un programa mal escrito nos podemos encontrar con errores de compilación, errores de ejecución, errores lógicos y defectos en la claridad del código. El siguiente programa está muy mal escrito. Señala y describe brevemente todos los errores y todos los defectos que veas. Deja claro si se trata de un error o un defecto, aunque no es necesario que especifiques si el error es de compilación, ejecución o lógico.

```

1 {$mode objfpc}{$H-}{$R+}{$T+}{$Q+}{$V+}{$D+}{$X-}{$warnings on}
2 program errores-1;
3
4 var
5     character: char;
6     marks_table: array[1..100] of real;
7     i: integer;
8
9 function valornumerico(carácter: char): integer;
10 begin
11     result := ord(character)-ord('0');
12 end;
13
14 procedure esmultiplode3(number: integer; result:boolean);
15 begin
16     result := ((number mod 3) = 0);
17 end;
18
19 function notapractnumerica(notapract: real; notaejerc: real): real;
20 begin
21     case notapract of
22         5.0:
23             notaejerc := result;
24         otherwise
25             result := 0.0;
26     end;
27 end;
28
29 function media( number1, number2: real): real;
30 begin
31     result := number1 + number2 / 2.0;
32 end;
33
34 begin
35     // Escribe True si la media de 5.0 y 7.0 es multiplo de 3
36     writeln(esmultiplode3(trunc(media(5.0, 7.0))));
37
38     // Rellena la tabla y escribe sus elementos en orden descendente
39     for i = 100 downto 0 do
40         marks_table[i] := notapractnumerica(5.0, 6.0 + real(i div 10));
41         write(marks_table[i], ' ');
42     writeln;
43 end.

```

Solución

1. Defecto. General: Mezcla de dos idiomas, español e inglés. Debería estar todo en inglés, o excepcionalmente, todo en español.
2. Error. Línea 2: El Identificador *Errores-1*, no es válido porque usa el carácter '-', que no está permitido.
3. Defecto. Línea 4: Variables globales. Estas variables deberían declararse después de los

subprogramas para que solo sean visibles en el cuerpo del programa principal.

4. Defecto. Línea 6: El *número mágico* 100 debería ser declarado como constante (como el resto de números del programa).
5. Error. Línea 9: El identificador *carácter* no es válido porque usa el carácter 'á', que no está permitido.
6. Error. Línea 11: Uso de la variable global *character* dentro de un *procedure*.
7. Error. Línea 14: El segundo parámetro, *result*, debe pasarse por referencia (*var result:boolean*).
8. Error. Línea 21: *Case* está usando una variable real para seleccionar una rama u otra. Esto no está permitido, solo pueden usarse tipos discretos.
9. Error. Línea 23: La asignación debe ser al revés, *result := notaejerc* ;
10. Error. Línea 31: La expresión para calcular la media es incorrecta, debería ser: $(number1 + number2) / 2.0$.
11. Error. Línea 36: La llamada a *esmultiplode3()* debe hacerse con 2 argumentos, falta el segundo.
12. Error. Línea 36: El argumento de un subprograma, en este caso *writeln*, no puede ser un procedimiento, porque un procedimiento no devuelve ningún valor.
13. Error. Línea 39: El operador de asignación es *:=*, no *=*. Por tanto, la asignación debería ser *i := 100*, no *i = 100*.
14. Error. Línea 39: El bucle for debería ser *100 downto 1*, no *100 downto 0*.
15. Error. Líneas 39-42: Falta el *begin-end* del *for*. Tal y como está escrito el programa, la sentencia *write* está fuera del bucle, a pesar de que la tabulación indica lo contrario.