

Nº Parcial:
Nombre:
CI:

Primer Parcial. Programación 1

Instituto de Computación
Mayo 2018

Observaciones:

- El parcial consta de 16 preguntas y un ejercicio de programación.
- Cada pregunta tiene una única opción correcta.
- Una respuesta correcta suma 2 puntos. Una respuesta incorrecta resta 0,5 puntos. Una pregunta sin respuesta no afecta el puntaje.
- El ejercicio de programación tiene un total de 8 puntos.
- Debe entregar la planilla con las respuestas junto con la letra y la resolución del ejercicio. Todo debe estar identificado con nombre y cédula.
- En todos los ejercicios se utiliza el lenguaje Pascal tal como fue dado en el curso (ej. circuito corto, case con else, etc.).

Ejercicio 1 Se considera la expresión

$(x < y) \text{ or } z$

Indique para qué declaración de tipos de las variables x , y y z , la expresión anterior será una expresión válida de tipo boolean:

- $x, y, z : \text{integer}$
- $z : \text{boolean}; x : \text{integer}; y : \text{real};$
- $x, z : \text{integer}; y : \text{boolean};$
- $x, y : \text{boolean}; z : \text{integer};$
- $x, y, z : \text{char};$

Respuesta al final: ¹

Ejercicio 2 Se consideran declarados dos subprogramas cuyos encabezados son:

```
procedure coco(var x,y : real);  
function papa(x : real; y : integer): integer;
```

Se consideran declaradas las variables:

```
var x,y : real;  
    i,j : integer;
```

Indicar cual de las siguientes instrucciones provoca un error en tiempo de compilación:

- `writeln(papa(papa(1,1),1))`
- `coco(x,y)`
- `coco(x,x)`
- `coco(x, x+y)`
- `x := papa(papa(i,j), trunc(x+y))`

Respuesta al final: ²

Ejercicio 3 Considere el siguiente fragmento, donde la variable x se supone declarada como integer:

```
if funciona(x,x,12) then  
  writeln('funciona')  
else  
  writeln(x)
```

¿Cuál de las siguientes opciones corresponde a un encabezado válido para la función funciona?:

- `function funciona(var x,y:integer; z:integer): integer;`
- `function funciona(var x,y,z: real): boolean;`
- `function funciona(var x : integer; y,z: real): boolean;`
- `function funciona(var x,y: real; z : integer): boolean;`
- `function funciona(x: real; var y,z: integer): boolean;`

Respuesta al final: ³

Ejercicio 4 Se considera el siguiente fragmento de programa:

```
type  
  rango = 1..5;  
  arrI = array [rango] of integer;  
  arrB = array [rango] of boolean;  
var  
  a : arrI;  
  b : arrB;  
  i,v : integer;  
begin  
  for i:= 1 to 5 do  
  begin  
    read(a[i]);  
    read(v);  
    b[i]:= a[i] > v  
  end;  
  i:= 6;  
  repeat  
    i:= i - 1;  
    write(b[i], ' ')  
  until i=1;  
end;
```

Indique cuál será la salida que produce el programa con la siguiente entrada: 1 2 4 3 5 4 3 4 2 1.

- TRUE TRUE TRUE FALSE FALSE
- FALSE FALSE FALSE FALSE FALSE
- FALSE FALSE TRUE FALSE TRUE
- 2 3 4 4 1
- TRUE FALSE TRUE TRUE FALSE

Respuesta al final: ⁴

Ejercicio 5 Considere la siguiente sentencia FOR:

```
for v := e1 to e2 do instrucción;
```

Determine qué afirmación es correcta:

- En *instrucción* es correcto asignarle un valor a v .
- Si $e_1 > e_2$ se produce un error.
- Luego de la ejecución completa del ciclo, se puede afirmar que la variable v queda con valor e_2 .
- La variable v debe ser de tipo integer.
- En *instrucción* se puede desplegar el valor de v (por ejemplo `write(v)`).

Respuesta al final: ⁵

Ejercicio 6 Se definen las siguientes variables:

```
var p,q : 0 .. 9;  
    a,b : '0'..'9';
```

Sabiendo que `ord('0') > 10`, indique cuál de las siguientes asignaciones produce un error de rangos (asuma que todas las variables están inicializadas):

- `p:= ord(a) - ord('0')`
- `a:= chr(ord('0') + (p + q) div 2)`
- `q:= sqr(ord(b)) mod 10`
- `a := chr(ord('0') + p mod 10)`
- `a := chr(p)`

Respuesta al final: ⁶

Ejercicio 7 Considere el siguiente código:

```

var suma, i, j : integer;
begin
  suma := 1;
  for i := 1 to 5 do
    for j := i+3 downto 5 do
      suma := suma + 1;
    writeln(suma)
  end
end

```

Indique la salida que despliega:

- a) 4
- b) 7
- c) 11
- d) 16
- e) 1

Respuesta al final: ⁷

Ejercicio 8 Sean las siguientes secuencias de instrucciones.

```

1)
read(a);
b := 0;
write (a, ' ', b);

```

```

2)
b := 0;
read(a);
write (a, ' ', b);

```

```

3)
read(a);
b := a;
write (a, ' ', b - a);

```

```

4)
read(a);
b := a;
write (b, ' ', a);

```

```

5)
read(a);
write (a, ' ', a);
b := 0;

```

Indique cuál de las siguientes afirmaciones es **incorrecta**.

- a) 1 y 2 despliegan los mismos valores para la misma entrada.
- b) 1 y 3 despliegan los mismos valores para la misma entrada.
- c) 4 y 5 despliegan los mismos valores para la misma entrada.
- d) 3 y 4 despliegan los mismos valores para la misma entrada.
- e) Para la entrada 0, todos los fragmentos despliegan los mismos valores.

Respuesta al final: ⁸

Ejercicio 9 Sean las siguientes expresiones:

- 1) $a + b / c$
- 2) $a + (b \text{ div } c)$
- 3) $a + (b / c)$
- 4) $(a + b) \text{ div } c$
- 5) $(a + b) / c$

Indique cuáles de ellas evalúan al mismo valor que la expresión $a + b \text{ div } c$. Asuma que todas las variables son enteras y están inicializadas con algún valor distinto de 0.

- a) Solamente 1.
- b) Solamente 1, 2 y 3.
- c) Solamente 2.
- d) Todas.
- e) Solamente 4.

Respuesta al final: ⁹

Ejercicio 10 Dado el siguiente programa:

```

program ejuntos;
const
  N = 3;
  pesos = '$';
var
  numeros: array[1..N] of integer;
  i, j: 1..N;
begin
  for i:=N downto 1 do
    numeros[i] := i;
  for i:=N downto 1 do
    begin
      for j:=numeros[i] to N do
        write(pesos);
      writeln;
    end;
  end.

```

¿Cuál será su salida?

- a) \$\$\$
 \$\$
 \$
- b) \$\$\$
- c) \$\$\$
 \$\$\$
 \$\$\$
- d) \$\$\$\$\$\$
- e) \$
 \$\$
 \$\$\$

Respuesta al final: ¹⁰

Ejercicio 11 Para el siguiente código, donde N es una constante mayor que 0 y las variables son de tipo Integer, diga cuál afirmación es correcta.

```

cont := 0;
x:= 0;
while (cont < N) or (x <> 5) do
begin
  read(x);
  if x = 5 then
    cont := 0
  else
    cont := cont + 1
end;
write(x)

```

- a) Se despliega el último valor leído.
- b) La iteración no termina nunca.
- c) La variable `cont` nunca va a tener el valor N.
- d) Se despliega el último valor leído distinto de 5.
- e) El cuerpo del `while` se ejecuta exactamente N veces.

Respuesta al final: ¹¹

Ejercicio 12 Siendo i, j variables de tipo entero, indique cuáles de los siguientes fragmentos de programa despliega 2 6 12.

```
1)
i:= 1; j:= 2;
while (i*j < 12) do
begin
  write (i*j, ' ');
  i:= j;
  j:= j+1;
end;
```

```
2)
j:= 2;
for i:= 1 to 3 do
begin
  write (i*j, ' ');
  j:= j+1;
end;
```

```
3)
i:=1; j:= 2;
repeat
  write (i*j, ' ');
  i:= j;
  j:= j+1;
until (i*j = 12);
```

- a) Solamente 1) y 2).
- b) Solamente 3).
- c) Solamente 2).
- d) Solamente 1) y 3).
- e) Solamente 2) y 3).

Respuesta al final: ¹²

Ejercicio 13 Considere los siguientes fragmentos de programa

```
var a,x : Integer;
begin
x:= 6;
read (a);
if (a > 3) and (a < 5) then
x := sqr (a)
else x := sqr (a) div 2;
writeln (x)
end.
```

```
var a,x : Integer;
begin
x:= 6;
read (a);
if (a > 3 ) then
if (a < 5) then
x := sqr (a)
else x := sqr (a) div 2;
writeln (x)
end.
```

Indique para cuál valor de entrada los dos fragmentos producen salida distinta:

- a) 3.
- b) 4.
- c) 5.
- d) 6.
- e) 7.

Respuesta al final: ¹³

Ejercicio 14 Dadas las siguientes declaraciones:

```
type Arreglo = array ['a'..'f'] of Boolean;
var a :Arreglo;
```

¿Cuál de las siguientes asignaciones es correcta ?

- a) a['f'] := a['g']
- b) a['b'] := a['h'] and a['b']
- c) a['a'] := not a['f']
- d) a[false] := true
- e) a['c'..'d'] := false..true

Respuesta al final: ¹⁴

Ejercicio 15 Indique cuál es la salida que produce el siguiente fragmento de programa. La variable a es entera.

```
a:= 4;
if a <= 10 then
  write('dale,');
if a <= 4 then
  case (a mod 3) of
    0: write('que');
    1: write('va');
  else
    write('che');
  end
else if sqr(a) <= 20 then
  write('viene')
```

- a) dale
- b) dale,va
- c) dale,que
- d) dale,che
- e) dale,va viene

Respuesta al final: ¹⁵

Ejercicio 16 Sea el siguiente fragmento de código donde x es de tipo integer:

```
x := 0;
if (x>0) and (10 mod x > 11) then
  write('IF1')
else
  write('ELSE1');
```

- a) El programa da error de compilación.
- b) El programa da error de ejecución.
- c) El programa escribirá IF1.
- d) El programa escribirá ELSE1.
- e) El programa no da error pero no despliega nada.

Respuesta al final: ¹⁶

Ejercicio de Resolución

- El puntaje máximo por este ejercicio es **8 puntos**.
- Este ejercicio debe resolverse en la hoja en blanco que se adjunta. Escriba su nombre completo y cédula en la hoja entregada.
- En este ejercicio se evaluará, además de la lógica correcta, la utilización de un buen estilo de programación de acuerdo a los criterios impartidos en el curso. De esta manera se tendrá en cuenta entre otros conceptos: indentación apropiada, correcta utilización de las estructuras de control, código elegante y legible, eficiencia de los algoritmos, pasaje correcto de parámetros, etcétera. Por razones prácticas no pedimos que incluya comentarios en el código que escriba.

Dadas las siguientes declaraciones:

```
const N = ... (* valor mayor que 0 *)
arr = array [1 .. N] of Integer;
```

Escribir una función

```
function siguiente(a:arr) : Boolean;
```

que reciba un arreglo de tipo `arr` como parámetro y devuelva `TRUE` si existe alguna celda `i` del arreglo que cumpla que: $a[i+1] = a[i]+1$. En caso contrario, la función debe retornar `FALSE`.

Ejemplos: En los siguientes ejemplos se considera $N = 6$.

- **Entrada:** |2|10|9|10|0|1|
Salida: TRUE
(dado que $a[4] = a[3] + 1$)
- **Entrada:** |-1|0|9|12|0|4|
Salida: TRUE
(dado que $a[2] = a[1] + 1$)
- **Entrada:** |2|10|9|12|-1|-2|
Salida: FALSE
(ya que no hay ninguna celda que cumpla la condición requerida)

Respuestas

¹b

²d

³c

⁴e

⁵e

⁶e

⁷c

⁸d

⁹c

¹⁰e

¹¹b

¹²c

¹³a

¹⁴c

¹⁵b

¹⁶d