



Estudis d'Informàtica, Multimèdia i Telecomunicació

Programació PAC1 — Document d'enunciats

Estudiant

Nom: Pere

Cognoms: Amengual Gomila

Exercici 1: Tipus i expressions [20%]

Tasca: Donades les següents definicions de tipus, constants i variables

```
const
  ICONSTA : enter = 5;
  RCONSTA : real = 12.6;
fconst
var
  i, j, k: enter;
  x, y, z: real;
  a, b, c: caràcter;
  t, u: cadena;
  b1, b2: booleà;
fvar
```

a) Avalueu les següents expressions amb potencials conflictes de tipus. Si l'expressió és correcta digueu quin seria el tipus resultant d'avaluar-la. Si l'expressió és incorrecta indiqueu l'error detectat. (*recordeu que l'operador = és de relació*):

i) `x * RCONSTA div j`
`real * real div enter`

incorrecta, error semàntic degut a que no es pot fer div d'un real amb un enter

ii) `realToInteger(y) div x`
`realToInteger(real) div real`
`enter div real`

incorrecta, error semàntic degut a que no es pot fer div d'un enter amb un real

iii) `b ≠ t o integerToReal(j div k) < y / 22.0`
`caràcter ≠ cadena o integerToReal (enter div enter) <`
`real / real`
`-error semàntic- o real < real`
`-error semàntic- o cert`

incorrecta, error semàntic degut a es barregen dos tipus de dades diferents amb l'operador ≠

iv) `charToCode(c) div realToInteger(x / 12.2)`
`enter div realToInteger(real / real)`
`enter div enter`

`enter`

correcta, el resultat seria del tipus `enter`

b) Creeu les següents expressions

- v) Expressió que digui si el codi del caràcter **a** té una valor superior a 100 i al valor de **x**

```
characterACodi('a') > 100 i characterACodi('a') > x
```

- vi) Expressió que digui si **b1** conté el resultat de saber si el contingut de la variable **z** correspon al producte de les variables **x** i **y**.

```
b1 = (z= x*y)
```

Exercici 2: Avaluació d'expressions [10%]

Tasca: Donades les definicions de constants, tipus i variables de l'exercici 1 i suposant que les variables tenen els valors següents:

```
i:=2, j:=4, k:=12, x:=0.75, y:=0.1, z:=1.0 a:='c', b:='Z',
c:='a', t:="aloha", u:="alelle", b1:=fals, b2:=cert
```

Avalueu el resultat de les expressions següents:

a) `realToInteger (RCONSTA / x) mod k`

```
realToInteger (12.6 / 0.75) mod 12
realToInteger (16.8) mod 12
16 mod 12
```

4

b) `k div j > i o t < u`

```
12 div 4 > 2 o "aloha" < "alelle"
3 > 2 o fals
cert o fals
```

cert

c) `realToInteger (z + x / y) div ICONSTA = k div j - i`

```
realToInteger (1.0 + 0.75/0.1) div 5 = 12 div 4 - 2
realToInteger (1.0 + 7.5) div 5      = 3 - 2
realToInteger (8.5) div 5            = 1
8 div 5 = 1
1 = 1
```

cert

d) `a ≠ 'a' = b2 = ((i + j) * i = k)`

```
'c' ≠ 'a' = cert = ((2+4)*2 = 12)
cert      = cert = (6 * 2 = 12)
cert      = cert = (12 = 12)
cert      = cert = cert
cert      = cert
```

cert

e) `integerToReal(ICONSTA) * RCONSTA / x + integerToReal (k)`

```
integerToReal(5) * 12.6 / 0.75 + integerToReal(12)
5.0 * 12.6 / 0.75 + 12.0
63.0 / 0.75 + 12.0
84.0 + 12.0
```

Exercici 3: Declaració de variables i especificació [10%]

Tasca: Declareu les variables necessàries i doneu les especificacions d'un algorisme que resolgui els problemes següents (no s'ha d'escriure l'algorisme):

*En concret, es demana que definiu les variables necessàries per representar les dades del problema (apartat **var...fvar** d'un algorisme) i que doneu la precondició i la postcondició.*

- a) Es desitja calcular a quina distancia ha recorregut un objecte que anava a velocitat constant (en km/h) coneixent les hores, minuts i segons que ha durat el moviment.

```
var
    velocitatKmh, hores, minuts, segons : enter;
    distanciaKm : real;

fvar

{ PRE: velocitatKmh ≥ 0 i hores ≥ 0 i minuts ≥ 0 i segons ≥ 0 }

    calcularDistància

{ POST: distanciaKm = enterAReal(velocitatKmh) *
  (enterAReal(hores) + enterAReal(minuts/60) +
  enterAReal(segons/3600)) }
```

- b) Que donada la superfície de la cara d'un cub, digui quin és el seu costat i quin el seu volum.

```
var
    superficie, costat, volum : real;

fvar

{ PRE: superficie > 0 }

    calcularDadesCub

{ POST: costat2 = superficie i costat > 0 i costat3 = volum }
```

c) Que donada una cadena de text recompti els espais en blanc i els guions que conté

```
var
```

```
    blancs, guions : enter;
```

```
    text: cadena;
```

```
fvar
```

```
{ PRE: es proporciona una cadena a avaluar }
```

```
    recompteCadena
```

```
{ POST: blancs és el nombre d'espais en blanc de la cadena text  
i guions ens el nombre de guions de la cadena text i  $\text{blancs} \geq 0$   
i  $\text{text} \geq 0$  }
```

Exercici 4: Comprensió i seguiment d'algorismes [20%]

Tasca: Donat el següent algorisme,

```
algorisme estrany
var
  n, c, result : enter;
fvar

  n := readInteger();
  mentre (n > 9) fer
    c := 0;
    mentre (n > 0) fer
      c := c + n mod 10;
      n := n div 10;
    fmentre
    n := c;
  fmentre
  si n > 5 llavors
    n := 9 - n;
  sino
    n := 2 * n;
  fsi
  writeInteger(n);
```

falgorisme

diguen què escriurà en el cas de les següents entrades

Entrada	Escriptura
17	1
21	6
706	8
12580	2
999999	0

aquesta part va descomponent la suma de les xifres que componen el nombre fins que arriba a un valor d'una sola xifra

aquesta part realitza una resta o una suma depenent del resultat anterior

Exercici 5: Problema, algorisme i programa en PHP [20%]

Tasca: Implementar una peculiar versió del joc de cartes de la carta més alta entre dos jugadors que trien cadascun una determinada quantitat de cartes (entre 1 i 5). La puntuació que s'assigna a cada jugador és el valor de la de la carta més alta que han tret segons la següent puntuació de les cartes:

Cartes	Valor
1 a 7	La puntuació nominal de la carta
Sota, Cavall i Rei	10 punts

La crida tindrà la forma

http://localhost/pac1_exer5.php?type=A&qtty=n on

- **type** : és un caràcter que indicarà quin tipus de carta es vol mostrar:

- **n** si es vol mostrar cartes napolitanes



- **e** si es vol mostrar cartes espanyoles



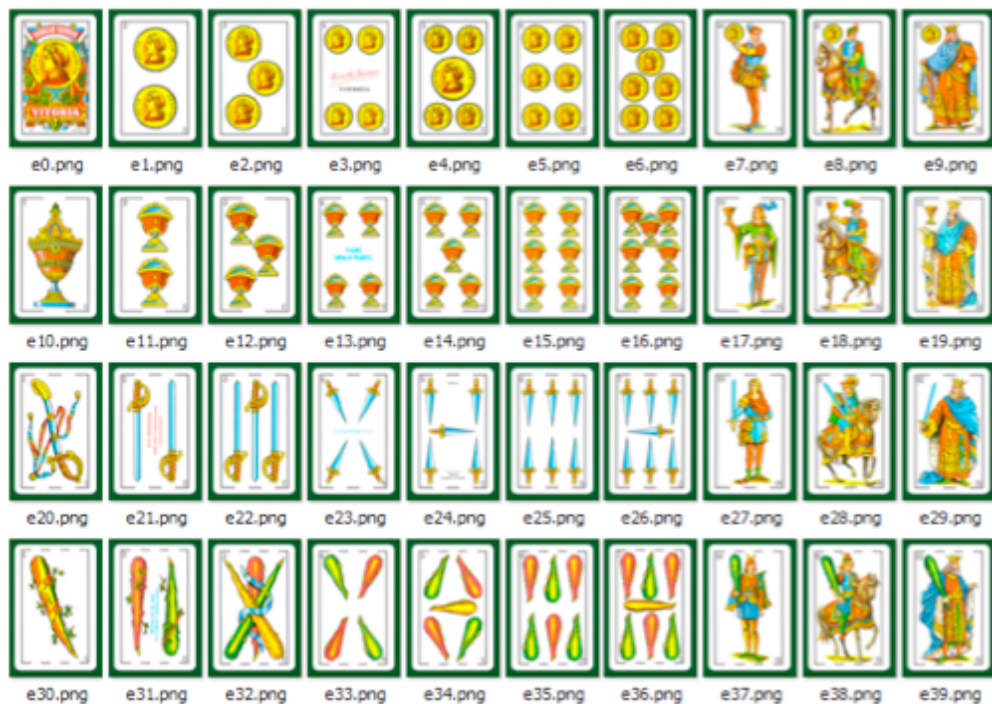
- **qtty** : la quantitat de cartes que tirarà cada jugador, un número entre **1** i **5**.

El programa haurà de complir els següents requisits:

- En el cas que **type** tingui un valor diferent als previstos, es mostrarà la baralla espanyola.
- En el cas que **qtty** tingui un valor fora de l'interval marcat pels números **1** i **5**, es tiraran 5 cartes per jugador.



- Al final mostrarà el missatge "**First player wins!**", "**Second player wins!**" o "**Dead head!**", segons hagi guanyador o hagin empatat.
- Podeu comprovar el funcionament a:
 - http://prog.uoc.edu/pac1/pac1_exer5.php?type=n&qtty=5
- Amb els fitxers de l'enunciat, es faciliten 80 fitxers d'imatge en format **png** corresponents a les cares de les cartes espanyoles i napolitanes.





- Cal que feu servir la plantilla **pac1_exer5_plantilla.php** per construir el programa en PHP i li canvieu el nom al fitxer a **pac1_exer5.php**

Indicacions

- No oblideu posar el nom de l'estudiant en la línia 12 de la plantilla.
- Els fitxers d'imatge s'han d'ubicar a la mateixa carpeta on estigui el fitxer php i no es poden canviar de nom.
- No es demana especialment en l'enunciat però es considerarà positivament que els caràcters **e** o **n** puguin ser introduïts indistintament en majúscula o minúscula.

A tenir en compte

- Cal observar els noms que han rebut els fitxers d'imatge:
 - El noms dels fitxers de les cartes espanyoles comencen per **e** i els de les napolitanes per **n**.
 - Un mateix número correspon a la mateixa carta en els dos jocs. Per exemple, les imatges **e15.png** i **n15.png** corresponen al 6 de copes
 - Cada desena de cartes correspon a un pal de la baralla i, en la desena, les cartes estan ordenades per valor creixent.
 - Per a les cartes amb valor entre **1** i **7**, la xifra del nom del fitxer que indica les unitats es correspon al valor de la carta disminuït en **1**. Per exemple, les imatges que acaben en **3** tenen el valor **4**.
- La sortida estàndard de PHP és l'escriptura d'un fitxer de text (html). Es pot construir el nom d'un fitxer d'imatge, a base de concatenar les seves parts. Per exemple, els codis equivalents donen com sortida **e12.png**

<pre>\$a = "e"; \$b = 12; print "\$a\$b.png";</pre>	<pre>\$a = "e"; \$b = 12; print \$a.\$b.".png";</pre>
---	---

- Per triar aleatòriament el número corresponent a una carta, es farà servir la funció **rand¹(min, max)** que rep dos nombres enters (**min** i **max**) i retorna un número enter entre ells, ambdós inclosos: Per al nostre cas, les crides a rand hauran de dur els paràmetres **0** i **39**, atès que els números de les cartes es troben en aquest interval.
- Per obtenir la darrera xifra d'un número, es farà servir l'operador **%** (mod²). Per exemple, si la variable **\$n** té el valor **12**, la instrucció **\$x = \$n % 10**; deixa en la variable **\$x** el valor **2**. Si **\$n** tingués el valor **34**, llavors emmagatzemaria en **\$x** el valor **4**, és a dir, el residu de dividir **\$n** per 10.
- Per obtenir el caràcter en minúscula corresponent a una lletra es pot fer servir la funció **strtolower(\$cadena)**³. Per exemple si la variable **\$a** té el valor "**T**", la instrucció **\$m = strtolower(\$a)**; deixaria en la variable **\$m** el caràcter "**t**". Si en **\$a** el valor emmagatzemat hagués estat "**t**", en **\$m** també s'hagués desat "**t**".

¹ <http://php.net/manual/es/function.rand.php>

² <http://www.php.net/manual/es/language.operators.arithmetic.php>

³ <http://www.php.net/manual/es/function strtolower.php>

Copieu aquí el codi de la vostra solució

```
<?php

$type = $_GET["type"];
$qtty = $_GET["qtty"];
$type = strtolower($type);

if ($type<>"n" and $type<>"e") $type="e";
if ($qtty<1 or $qtty>5) $qtty=5;

$first= array();
$altafirst=0;
print "<h4>First Player</h4> ";

for ($i=1; $i<=$qtty; $i++) {
    $first[$i]= rand(0, 39);
    $mostra=$first[$i];
    $valor=$first[$i] % 10 +1;
    print "<img src='$type$mostra.png'> ";
    if ( ($valor) >= $altafirst) $altafirst = ($valor) ;
}

if ($altafirst>=8) $altafirst=10;
print "<br> Max score: $altafirst; ";
echo "<br>";

$second= array();
$altasecond=0;
print "<h4>Second Player</h4> ";

for ($i=1; $i<=$qtty; $i++) {
    $second[$i]= rand(0, 39);
    $mostra=$second[$i];
    $valor=$second[$i] % 10 +1;
    print "<img src='$type$mostra.png'> ";
    if ( ($valor) >= $altasecond) $altasecond = ($valor) ;
}

if ($altasecond>=8) $altasecond=10;
print "<br> Max score: $altasecond; ";
echo "<br>";

if ($altafirst > $altasecond) echo "<h3>First player wins!</h3>";
if ($altafirst < $altasecond) echo "<h3>Second player wins!</h3>";
if ($altafirst == $altasecond) echo "<h3>Dead head!</h3>";

?>
```

Exercici 6: Problema, algorisme i programa en PHP [20%]

Tasca: Simular una sessió d'un jugador en una màquina escurabutxaques.

<p>El programa rep a la crida una dada, de nom capital, un número enter que indica de quants diners es disposa quan es va a començar a jugar.</p> <p>La crida ser</p> <p><code>http://localhost/pac1_exer6.php?capital=3</code></p> <p>El joc implica una quantitat variable de partides. S'aturarà bé quan el jugador es quedi sense capital o bé quan hagi guanyat i sigui igual o ultrapassi el doble del capital inicial (en la imatge d'exemple el capital inicial és 1 i el joc finalitza perquè el capital del jugador arriba a 2.5)</p> <p>Cada aposta té el valor determinat per la constant que es definirà amb el nom BET i el valor 0.5.</p> <p>En iniciar el joc es mostrarà el capital inicial i, a cada tirada el número de jugada, les tres figures obtingudes, el guany de la tirada i el capital total després d'haver pagat l'aposta i, en el seu cas, de cobrar els guanys.</p> <p>En finalitzar el joc es mostrarà un missatge que indiqui el motiu, per haver perdut tots els diners o per haver guanyat massa.</p>	
---	--

La taula de premis serà la següent:

Combinació	Premi
Tres cops la figura	100 cops l'aposta
Tres figures iguals (llevat de)	6 cops l'aposta
Dos cops la figura	4 cops l'aposta
Dos cops una altra figura	2 cops l'aposta
Tres figures diferents	0

Es pot comprovar el funcionament a:

http://prog.uoc.edu/pac1/pac1_exer6.php?capital=3

El programa s'escriurà a la plantilla **pac1_exer6_plantilla.php** que es canviarà de nom a **pac1_exer6.php** pel lliurament


Indicacions:

- No oblideu posar el nom de l'estudiant en la línia 12 de la plantilla.
- Es faciliten les 9 imatges per mostrar les figures



- Els fitxers d'imatge s'han d'ubicar a la mateixa carpeta on estigui el fitxer php i no es poden canviar de nom.
- Les tres imatges que han de sortir en cada tirada es triaran a l'atzar.

A tenir en compte

- Per declarar una variable amb nom, en php es fa servir la funció **define(\$name,\$value)**⁴. Per exemple `define("PI", 3.14);` crea la constant **PI** a la qual se li associa el valor **3.14**
- A diferència de l'exercici anterior, els noms dels fitxers no segueixen cap norma especial.
- La funció **rand (min, max)**⁵ que ja s'ha fet servir en l'exercici anterior és molt útil per generar valors aleatoris, però només ho fa amb números, de manera que s'aconsella fer servir rand per generar números aleatoris entre 1 i 9 i programar el segment de programa que triï la imatge a mostrar en funció del número (per exemple 1→banana, 2→bell, etc. Si es dóna a la figura  el valor més alt, es poden simplificar algunes condicions de les que regeixin les composicions d'accions.
- No oblideu que a cada tirada el capital s'ha de disminuir en el valor de l'aposta, independentment de si després es guanya o no.
- A les proves de programació s'aconsella emprar un capital petit (2, 3) per tal d'evitar l'ús enutjós del scroll del navegador per arribar al missatge final.

⁴ <http://www.php.net/manual/es/function.define.php>

⁵ <http://php.net/manual/es/function.rand.php>

Copieu aquí el codi de la vostra solució

```
<?php

define('BET', 0.5);
$capital = $_GET["capital"];
$capitalinicial = $capital;
$gamenum = 0;
$figura=array("banana","bell","grape","lemon","orange","peach","plum","watermelon",
"seven");
$obtingudes=array();

echo "<h3>Initial Capital: $capitalinicial </h3>";
echo "<h3>Actual Capital: $capital </h3>";

while ($capital>0 and $capital<=$capitalinicial * 2) {

    $guany=0;
    $gamenum=++$gamenum;
    echo "<h3>Game $gamenum </h3>";
    $capital=$capital - BET;
    for ($i=0; $i<=2; $i++) {
        $obtingudes[$i]=rand(0,8);
        /*print "$obtingudes[$i]";*/
        $mostrar=$obtingudes[$i];
        print "<img src='$figura[$mostrar].png'>";
    }

    if (($obtingudes[0]==$obtingudes[1] and $obtingudes[0]<>8) or
($obtingudes[0]==$obtingudes[2] and $obtingudes[0]<>8) or
($obtingudes[1]==$obtingudes[2]) and $obtingudes[1]<>8) {
        /*echo "dues figures iguals ";*/
        $guany=BET*2;
    }

    if (($obtingudes[0]==$obtingudes[1] and $obtingudes[0]==8) or
($obtingudes[0]==$obtingudes[2] and $obtingudes[0]==8) or
($obtingudes[1]==$obtingudes[2]) and $obtingudes[1]==8) {
        /*echo "dos 7 iguals ";*/
        $guany=BET*4;
    }

    if ($obtingudes[0]==$obtingudes[1] and $obtingudes[0]==$obtingudes[2] and
$obtingudes[0]==8) {
        /*echo "super 7 ";*/
        $guany=BET*100;
    }

    if ($obtingudes[0]==$obtingudes[1] and $obtingudes[0]==$obtingudes[2] and
$obtingudes[0]<>8) {
        /*echo "tres iguals ";*/
        $guany=BET*6;
    }

    $capital=$capital+$guany;
    echo "<br>Gain $guany Actual capital = $capital";

}

if ($capital==0) {
    print "<h3>You lost too much</h3>";
}
else {
    print "<h3>You won too much</h3>";
}

?>
```