



## EJERCICIOS RESUELTOS: ESTRUCTURAS DE PROGRAMACIÓN COMPLEJAS - SOLUCIONES

### Problema 1

Defina una función que se llame “DatosPaciente” que defina una estructura de datos que permita describir la Tabla1. Inserte los valores que contiene dicha tabla, nombre de las columnas y filas, tal y como aparece en la Tabla1.

En la misma función, inserte dos columnas, una con el sexo y otra con la Edad. La tabla resultante debería ser como la tabla 2.

	Altura	Peso
Ana	1.65	65
Pepe	1.74	80
Nacho	1.70	77
Bea	1.55	60
Gema	1.75	66
Alba	1.58	65

Tabla1. Altura y Peso

	Altura	Peso	Sexo	Edad
Ana	1.65	65	M	23
Pepe	1.74	80	H	43
Nacho	1.70	77	H	34
Bea	1.55	60	M	43
Gema	1.75	66	M	45
Alba	1.58	65	M	54

Tabla2. Altura, Peso, Sexo y Edad

En la misma función, inserte dos columnas más, una con el cálculo del índice de masa corporal y en otra con el “Nivel Peso”. EL IMC se calcula:  $IMC = \text{peso [kg]} / \text{estatura [m}^2\text{]}$ . Para la columna “Nivel Peso”, los rangos se calculan en función del IMC, si el IMC es: Menor de 18.5 “Bajo peso”; Entre 18.5-24.9 “Normal”; Entre 25.0 – 29.0 “Sobrepeso”; Mas de 30 “Obeso”

	Altura	Peso	Sexo	Edad	IMC	Nivel Peso
Ana	1.65	65	M	23	23.88	Normal
Pepe	1.74	80	H	43	26.42	Sobrepeso
Nacho	1.70	77	H	34	26.64	Sobrepeso
Bea	1.55	60	M	43	24.97	Normal
Gema	1.75	66	M	45	21.55	Normal
Alba	1.58	65	M	54	26.04	Sobrepeso

Tabla2. Altura, Peso, Sexo, Edad, IMC y Nivel Peso

```

DatosPacientes<-function()
{
  paciente=matrix(c(1.65,1.74,1.70,1.55,1.75,1.58,65,80,77,60,66,65),ncol=2)
  colnames(paciente)=c("Altura","Peso")
  rownames(paciente)=c("Ana","Pepe","Nacho","Bea","Gema","Alba")
  print(paciente)
  paciente=data.frame(paciente)
  print(paciente)
  sexo=c("M","H","H","M","M","M")
  paciente =cbind(paciente,sexo)
  print(paciente)
  edad=c(23,43,34,43,45,54)
  paciente =cbind(paciente,edad)

  print(paciente)
  for(i in 1:nrow(paciente))
  {
    print(i)
    paciente[i,"IMC"]=(paciente[i,"Peso"]/(paciente[i,"Altura"]^2))
    print(paciente)
    if (paciente[i, "IMC"] < 18.5)
      paciente[i,"NivelPeso"]="Bajo Peso"
    else
      if ((paciente[i, "IMC"] >= 18.5) & (paciente[i, "IMC"] < 24.9))
        paciente[i,"NivelPeso"]="Normal"
      else
        if ((paciente[i, "IMC"] >= 25) & (paciente[i, "IMC"] < 29))
          paciente[i,"NivelPeso"]="Sobrepeso"
        else
          paciente[i,"NivelPeso"]="Obeso"
    print(paciente)
  }
}

```

## Problema 2

Escriba una función que lea una matriz A y compruebe: si la posición (suma fila-columa) es impar comprobar si el valor es impar, si coincide que ponga un 0 si no que ponga un -1; los mismo con I sustituya los valores iguales por un -1 en la matriz B. **Realizarlo con for, while y repeat**

$$A = \begin{pmatrix} 3 & 3 & 4 \\ 6 & 4 & 8 \end{pmatrix}$$

$$\text{Resultado: } A = \begin{pmatrix} -1 & 0 & 0 \\ -1 & 0 & -1 \end{pmatrix}$$

Fila:1; Columna.1-> 1+1 =2 (Par)  
 Valor de A[1,1] = 3 (Impar)  
 Como son distintos: -1

```
iguales<-function()
{
  A=matrix(c(3,6,3,4,4,8),2,3)
  for(i in 1:nrow(A))
  for (j in 1:ncol(A))
  if (((i+j)%2)==0)
  if((A[i,j]%2)==0)
    A[i,j]=0
  else
    A[i,j]=-1
  else
  if((A[i,j]%2)!=0)
    A[i,j]=0
  else
    A[i,j]=-1
  print(A)
}
```

```
igualesWhile<-function()
{
  A=matrix(c(3,6,3,4,4,8),2,3)
  i=1
  while(i<=nrow(A))
  {
    j=1
    while(j<=ncol(A))
    {
      if (((i+j)%2)==0)
      if((A[i,j]%2)==0)
        A[i,j]=0
      else
        A[i,j]=-1
      else
      if((A[i,j]%2)!=0)
        A[i,j]=0
      else
        A[i,j]=-1
      j=j+1
    }
    i=i+1
  }
  print(A)
}
```

```
igualesRepeat<-function()
{
  A=matrix(c(3,6,3,4,4,8),2,3)
  i=1
  repeat {
    j=1
    repeat
    {
      if (((i+j)%2)==0)
      {
        if((A[i,j]%2)==0)
          A[i,j]=0
        else
          A[i,j]=-1
      }
      else
      {
        if((A[i,j]%2)!=0)
          A[i,j]=0
        else
          A[i,j]=-1
      }
      j=j+1
      if(j>ncol(A))
        break
    }
    i=i+1
    if(i>nrow(A))
      break
  }
  print(A)
}
```