

## Transmisión de Datos 2011/12

- **Ejercicio 3 – Resolución Detallada: Pseudo-código Codificación Huffman binaria**

Escribir el pseudo-código de: el algoritmo de codificación Huffman (para crear el código), un codificador Huffman (con código –tabla- único por secuencia), y un decodificador Huffman (con código –tabla- único por secuencia).

➤ *ALGORITMO DE CODIFICACIÓN HUFFMAN*

*funcion [codigo]= Huffman ( P )*

**P** = vector de probabilidades como entrada  
**codigo** = tabla vacia inicialmente

Si (longitud de P < 2) OR (Sum(P) != 1)  
error y sale  
fin

**P\_ordenada** = ordena de mayor a menor(P)

Si la longitud es 2  
asigna 0 y 1 según criterio  
**codigo** = almacena probabilidad y codigo asignado en columnas adecuadas  
en caso contrario  
**P\_ordenada** = sustituye las dos 2 probabilidades menores por la suma de estas

**codigo** = almacena probabilidad y codigo asignado para cada probabilidad de las sumadas, en sus respectivas columnas

**Huffman(P\_ordenada)** comienza de nuevo con el nuevo vector ordenado  
fin

➤ *CODIFICADOR HUFFMAN*

*funcion [secuencia]= HuffmanCodificador ( informacion )*

**P** = calcula vector de probabilidades de *informacion*  
**Sum**= suma de las probabilidades es cero inicialmente

para i=1 hasta fin del vector de **P**  
**sum=sum+P(i)**

```
si sum>1
    error y sale
fin
fin
```

```
si sum es < 1
    error y sale
fin
```

**código** dado por: huffman de P

**código (tabla)** = código + columna de símbolos asociados a cada probabilidad

**secuencia** = vacía

```
para i=1 hasta fin del vector de informacion
```

```
    símbolo = busca en código el símbolo correspondiente a informacion (i)
```

```
    secuencia = [secuencia  símbolo]
```

```
fin
```

➤ *DECODIFICADOR HUFFMAN*

*funcion [informacion]= HuffmanRecursivo ( secuencia, código )*

almacenado = vacío inicialmente, para almacenar bits recibidos

informacion = vacía inicialmente

```
para j=1 hasta fin de secuencia
```

```
    almacenado = [almacenado  secuencia(j)]
```

```
    para i=1 hasta fin de la columna de códigos de código
```

```
        si almacenado = código(columna de códigos,i)
```

```
            informacion = [informacion  código(columna de símbolos,i)]
```

```
        fin
```

```
    fin
```

```
fin
```