

1. Modifique el código del algoritmo que resuelve el problema de la moneda utilizando backtracking, de manera que devuelva qué monedas se utilizan, en vez de sólo la cantidad.

```
type Monedero = tuple
    fst: nat          {- cantidad de monedas usadas -}
    snd: List of nat  {- denominaciones de las monedas que se usan para pagar -}
end tuple

fun cambio(d: array[1..n] of nat, i: nat, j: nat) ret res: Monedero
    var aux1,aux2: Monedero

    if j = 0 then
        res.fst := 0
        res.snd := empty_list()
    else if i := 0 then
        res.fst := ∞
        res.snd := empty_list()
    else if d[i] > j then
        res := cambio(d,i-1,j)
    else
        aux1 := cambio(d,i-1,j)
        aux2 := cambio(d,i,j-d[i])

        if aux1.fst < (1 + aux2.fst) then {- si aux1 usa menos monedas que aux2, elijo aux1 -}
            res := aux1
        else {- si aux2 usa menos monedas que aux1, elijo aux2 -}
            addr(aux2.snd,d[i])
            res.fst := 1 + aux2.fst
            res.snd := aux2.snd
        fi
    fi
fi
end fun
```