

```

(*El siguiente modulo devuelve los prefijos*)

Pref[x_] := Module[{res, i, w},
    (*módulo

    res = {{}};

    (*Bucle amb Take i Append*)
    (*toma      *añade

    For[i = 1, i ≤ Length[x], i++,
    (*para cada      *longitud
        w = Take[x, {1, i}];
        (*toma
        AppendTo[res, w];
        (*añade al final
    ];
    Return [res]
    (*retorna
]


```

---

```

Ex1[M_] := Module[{est, alf, trans, i, aux, j, cad, resposta},
    (*módulo

    (*Conjunto de Estados del arbol 'est'*)
    est = {{}};

    For[i = 1, i ≤ Length[M], i++,
    (*para cada      *longitud
        aux = Pref[M[[i]]];
        est = Join[est, aux];
        (*junta
    ];
    est = Union[est];
    (*unión

    (*Conjunto de simbolos 'alf'
    Flatten el que fa es fer que la llista de
    (*aplana
    llistes se convertisca en una unica llista tot seguit.
    Per exemple:{{a},{b,b},{a,a,a}} al aplicarli Flatten torna:
    (*aplana
    {a,b,b,a,a,a} desfa les llistes i
    torna tots els elements en una unica llista. El
    union se fa per a llevar elements repetits...
    aixi que tindrem els simbols del alfabet nomes. *)
    alf = Union[Flatten[M]];
    (*unión      *aplana

    (*Conjunto de transiciones del arbol 'trans'*)
    trans = {};

    For[i = 1, i ≤ Length[est], i++,
    (*para cada      *longitud

```

```

(*Para cada longitud*)
For[j = 1, j ≤ Length[alf], j++,
  (*para cada longitud*)
  (*Concatena x amb a. 'xa' es el
  que apareix al final del full 1 de 4 de la practica*)
  cad = Append[est[[i]], alf[[j]]];
  (*añade*)
  (*Mirem si la cadena cad pertany a Q,
  que per a nosaltres es un estat 'est' Es el que possa al
  final del full 1 de 4*)
  If[MemberQ[est, cad], AppendTo[trans, {est[[i]], alf[[j]], cad}]];
  (*si ¿contenido en? añade al final*)
];
];
(*Guardar el automat en una variable resposta.
El estado inicial del automada
es el vacio lambda y el final es la propia M.
Return[est, alf, trans, {}, M]*)
(*retorna*)
resposta = Join[{est}, {alf}, {trans}, {{}}, {M}];
(*junta*)

Return[resposta];
(*retorna*)
]

M = {{a}, {b, b}, {a, a, a}, {a, a, b}, {a, b, b},
      {a, a, a, b}, {a, a, b, a}, {a, a, b, a, b}, {a, a, b, b, b}};

Ex1[M]
{{{ {}, {a}, {b}, {a, a}, {a, b}, {b, b}, {a, a, a}, {a, a, b}, {a, b, b},
  {a, a, a, b}, {a, a, b, a}, {a, a, b, b}, {a, a, b, a, b}, {a, a, b, b, b}},
  {a, b}, {{{}, a, {a}}, {{}, b, {b}}, {{a}, a, {a, a}}, {{a}, b, {a, b}},
  {{b}, b, {b, b}}, {{a, a}, a, {a, a, a}}, {{a, a}, b, {a, a, b}},
  {{a, b}, b, {a, b, b}}, {{a, a, a}, b, {a, a, a, b}},
  {{a, a, b}, a, {a, a, b, a}}, {{a, a, b}, b, {a, a, b, b}},
  {{a, a, b, a}, b, {a, a, b, a, b}}, {{a, a, b, b}, b, {a, a, b, b, b}}},
  {}, {{{a}, {b, b}, {a, a, a}, {a, a, b}, {a, b, b}, {a, a, a, b},
  {a, a, b, a}, {a, a, b, a, b}, {a, a, b, b, b}}}

```

---

(\*L'exercici2 partix de una copia exacta de l'exercici 1,  
per s'afegeix un bucle que se mantinga en l'estat cadena  
buida. El codi que veuras es una copia del primer pero sense  
tants comentaris llargs. Qualsevol dubte mirar el Ex1\*)

```

Ex2[M_] := Module[{est, alf, trans, i, aux, j, cad, resposta},
  (*módulo*)
  (*Estados del arbol 'est'*)
  est = {{}};

```

```

For[i = 1, i ≤ Length[M], i++,
  para cada longitud
    aux = Pref[M[[i]]];
    est = Join[est, aux];
    junta
];
est = Union[est];
unión

(*Simbolos 'alf'*)
alf = Union[Flatten[M]];
unión aplana

(*Transiciones del arbol 'trans'*)
trans = {};
For[i = 1, i ≤ Length[alf], i++,
  para cada longitud
    (*El següent AppendTo es la RESPOSTA DEL EXERCICICI 2. Es lo unic
      añade al final
      que canvia respecte al Ex1 i es un bucle en l'estat inicial.*)
    AppendTo[trans, {{}, alf[[i]], {}]];
    añade al final
];

For[i = 1, i ≤ Length[est], i++,
  para cada longitud
    For[j = 1, j ≤ Length[alf], j++,
      para cada longitud

        (*Concatena x amb a. 'xa' es el
          que apareix al final del full 1 de 4 de la practica*)
        cad = Append[est[[i]], alf[[j]]];
        añade

        (*Mirem si la cadena cad pertany a Q,
          que per a nosaltres es un estat 'est' Es el que possa al
          final del full 1 de 4*)
        If[MemberQ[est, cad], AppendTo[trans, {est[[i]], alf[[j]], cad}]];
        si ¿contenido en? añade al final
      ];
    ];

(*Guardar el automat en una variable resposta.
  El estado inicial del automada
  es el vacio lambda y el final es la propia M.
  Return[est, alf, trans, {}, M]*)
retorna

resposta = Join[{est}, {alf}, {trans}, {{}}, {M}];
junta

```

```
Return[resposta];
|retorna
]
```

**Ex2 [M]**

```
{{ {}, {a}, {b}, {a, a}, {a, b}, {b, b}, {a, a, a}, {a, a, b}, {a, b, b},
  {a, a, a, b}, {a, a, b, a}, {a, a, b, b}, {a, a, b, a, b}, {a, a, b, b, b}},
{a, b}, {{ {}, a, {}}, {{ }, b, {}}, {{ }, a, {a}}, {{ }, b, {b}},
  {{a}, a, {a, a}}, {{a}, b, {a, b}}, {{b}, b, {b, b}}, {{a, a}, a, {a, a, a}},
  {{a, a}, b, {a, a, b}}, {{a, b}, b, {a, b, b}}, {{a, a, a}, b, {a, a, a, b}},
  {{a, a, b}, a, {a, a, b, a}}, {{a, a, b}, b, {a, a, b, b}},
  {{a, a, b, a}, b, {a, a, b, a, b}}, {{a, a, b, b}, b, {a, a, b, b, b}}},
{ }, {{a}, {b, b}, {a, a, a}, {a, a, b}, {a, b, b}, {a, a, a, b},
  {a, a, b, a}, {a, a, b, a, b}, {a, a, b, b, b}}}
```

---

**(\*Exercici 3.\*)**

---

```

Ex3[m_, x_] :=
Module[{aut, estado, estadest, trans, fin, i, j, t, aux, respuesta, aux2, k},
  (*módulo*)
  respuesta = {};
  aut = Ex2[m];
  estado = {aut[[4]]};
  trans = aut[[3]];
  fin = aut[[5]];

  For[i = 1, i ≤ Length[x], i++,
    (*para cada longitud*)
    estadest = {};
    For[j = 1, j ≤ Length[estado], j++,
      (*longitud*)
      aux = Cases[trans, {estado[[j]], x[[i]], _}];
      (*casos*)
      For[k = 1, k ≤ Length[aux], k++,
        (*longitud*)
        AppendTo[estadest, aux[[k, 3]]];
        (*añade al final*)
      ];
    ];
    estado = estadest;
    aux2 = Intersection[fin, estado];
    (*intersección*)
    For[k = 1, k ≤ Length[aux2], k++,
      (*longitud*)
      AppendTo[respuesta, {i - Length[aux2[[k]]] + 1, aux2[[k]], i}];
      (*longitud*)
    ];
  ];
  Return[respuesta];
  (*retorna*)
];

M3 = {{b, b}, {a, b, b, b}, {b, b, a, b}, {a, a, a, a}};

x = {b, a, b, a, a, b, b, a, b, b, b,
     a, b, b, a, b, a, a, a, a, b, b, a, a, b, b, a, b, a};

Ex3[M3, x]
{{6, {b, b}, 7}, {6, {b, b, a, b}, 9}, {9, {b, b}, 10}, {10, {b, b}, 11},
 {8, {a, b, b, b}, 11}, {10, {b, b, a, b}, 13}, {13, {b, b}, 14},
 {13, {b, b, a, b}, 16}, {17, {a, a, a, a}, 20}, {18, {a, a, a, a}, 21},
 {22, {b, b}, 23}, {26, {b, b}, 27}, {26, {b, b, a, b}, 29}}

```