

Ecrire une procédure $truc(t)$ qui renvoie "vrai" si t vérifie la propriété Π , "faux" sinon.

On suppose que la propriété Π (ou sa négation) peut s'écrire sous la forme

$$\forall x \in \dots, \Psi(x)$$

```

truc :=proc(t)
....
if ... then
    return "faux"    { On a trouvé un x qui ne vérifie pas  $\Psi$  }
...
fi;
return "vrai";      { ... puisqu'on n'a pas trouvé de x qui ne vérifie pas  $\Psi$  }
end;

```

Exemples :

1. t est un tableau d'entiers, indexé de 0 à $n - 1$ et Π est la propriété : $\exists i \in \llbracket 0, n - 1 \rrbracket, t[i] > 15$, donc sa négation est

$$\forall i \in \llbracket 0, n - 1 \rrbracket, \Psi(i) \text{ où } \Psi(i) = "t[i] < 16".$$

```

truc :=proc(t)
global n;
local i;
for i from 0 to n-1 do
    if t[i]>15 then
        return "vrai"
    fi;
od;
return "faux";
end;

```

2. Avec 2 quantificateurs : t est un tableau d'entiers de $E_n = \llbracket 0, n - 1 \rrbracket$, indexé de 0 à $n - 1$ identifié à une application de E_n dans E_n et Π est la propriété : $\forall i \in \llbracket 0, n - 1 \rrbracket, \exists k \in \llbracket 0, n - 1 \rrbracket, t^k[i] > 15$, ie :

$$\forall i \in \llbracket 0, n - 1 \rrbracket, \Psi(i) \text{ où } \Psi(i) = "\exists k \in \llbracket 0, n - 1 \rrbracket, t^k[i] > 15".$$

On commence par écrire la procédure $chose(i)$ qui renvoie "vrai" si i vérifie Ψ , "faux" sinon :

```

chose :=proc(i)
global n;
local k,iItere;
iItere :=i;
for k from 0 to n-1 do
    if iItere>15 then
        return "vrai"
    fi;
    iItere :=t[iItere]
od;
return "faux";
end;

```

puis on l'appelle dans $truc$:

```

truc :=proc(t)
global n;
local i;
for i from 0 to n-1 do
    if chose(i)="faux" then
        return "faux"
    fi;
od;
return "vrai";
end;

```