

0.1 Des algorithmes encadrés

La version non étoilée de l'environnement `algo` ajoute un cadre, comme ci-dessous, afin de rendre plus visibles les algorithmes.

Donnée : $n \in \mathbb{N}^*$

Résultat : $\sum_{i=1}^n i$

Actions

```
|  $s \leftarrow 0$   
| for  $i$  allant de 1 à  $n$  do  
|   |  $s \leftarrow s + i$   
| return  $s$ 
```

Algorithme 1 : Un truc bidon

Le code utilisé pour obtenir le rendu précédent est le suivant où sont utilisées certaines des macros additionnelles proposées par `lyalgo` (voir la section ??) où les macros `\NNs` et `\dsum` viennent du package `lymath`.

Code $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$

```
\begin{algo}  
  \caption{Un truc bidon}  
  \Data{$n$ \in \NNs$}  
  \addalgorithmbank  
  \Result{$\dsum_{i = 1}^n i$}  
  \Actions{  
    $s$ \Store 0$  
    \\  
    \ForRange{$i$}{$1$}{$n$}{  
      $s$ \Store $s + i$  
    }  
    \Return{$s$}  
  }  
\end{algo}
```

L'environnement `algo` propose un argument optionnel pour indiquer la largeur relativement à celle des lignes. Ainsi via `\begin{algo}[.45] ... \end{algo}`, on obtient la version suivante bien moins large de l'algorithme précédent.

Donnée : $n \in \mathbb{N}^*$

Résultat : $\sum_{i=1}^n i$

Actions

```
|  $s \leftarrow 0$   
| for  $i$  allant de 1 à  $n$  do  
|   |  $s \leftarrow s + i$   
| return  $s$ 
```

Algorithme 2 : Un truc bidon

On peut utiliser un environnement `multicols` pour un effet sympa.

Donnée : $n \in \mathbb{N}^*$

Résultat : $\sum_{i=1}^n i$

Actions

```
|  $s \leftarrow 0$   
| for  $i$  allant de 1 à  $n$  do  
|   |  $s \leftarrow s + i$   
| return  $s$ 
```

Algorithme 3 : Un truc bidon

Donnée : $n \in \mathbb{N}^*$

Résultat : $\sum_{i=1}^n i$

Actions

```
|  $s \leftarrow 0$   
| for  $i$  allant de 1 à  $n$  do  
|   |  $s \leftarrow s + i$   
| return  $s$ 
```

Algorithme 4 : Un truc bidon