

Recherche dichotomique

Polytech Nice-Sophia
Département informatique
Preuves de programmes
Sylvain Lippi

On propose deux versions de la recherche dichotomique dans un tableau t d'entiers triés. On fait la recherche entre les indices 1 et n on ne fait aucune hypothèse sur $t[0]$. Si la recherche réussit on rend l'indice correspondant et zéro sinon.

Exercice 1 *Une version simple Une première version « naturelle » pourrait être.*

```
1  int bsearch(int t[], int n, int v)
2  {
3      int l=1, u=n;
4
5      while (l<=u)
6      {
7          int m=(l+u)/2;
8
9          if (v<t[m]) u=m-1;
10         else if (v>t[m]) l=m+1;
11         else return m;
12     }
13     return 0;
14 }
```

Donner une spécification complète de cette fonction et la prouver avec `frama-c -jessie`.

Exercice 2 *Modification de l'intervalle de recherche*

1. On souhaite faire la recherche dichotomique entre les indices 0 et $n-1$. Modifier la version précédente et prouver sa correction.
2. Que se passe-t-il si l'on supprime la directive `#pragma JessieIntegerModel(math)` ?

Exercice 3 *Une version améliorée On propose une version légèrement modifiée qui rend le plus petit indice en cas de succès. Prouver cette propriété supplémentaire.*

```
1  int ibsearch(int t[], int n, int e)
2  {
3      int l=0, u=n;
4
5      while (l+1<u)
6      {
7          int m=(l+u)/2;
8          if (t[m]<e) l=m; else u=m;
9      }
10     if (t[u]==e) return u;
11     else return 0;
12 }
```