

```
1 -module(recursion).
2 -compile(export_all).
3
4 bump([]) -> [];
5 bump([Head|Tail]) -> [Head+1|bump(Tail)].
6
7 average(List) -> sum(List) / len(List).
8
9 sum([]) -> 0;
10 sum([Head|Tail]) -> Head + sum(Tail).
11
12 len([]) -> 0;
13 len([_|Tail]) -> 1 + len(Tail).
14
15 even([]) -> [];
16 even([Head|Tail]) when Head rem 2 == 0 -> [Head|even(Tail)];
17 even([_|Tail]) -> even(Tail).
18
```

Find:  ☒ Next ☐ Previous ☐ Match CaseGoto Line: 

Step

Next

Continue

Finish

Where

Up

Down

Evaluator: 

Name

Value

Tail

[3,4,5]

Head

2