

1. Complete the following definition of alternate, which interleaves two lists into one, alternating between elements taken from the first list and elements from the second.

1. Fixpoint `alternate (l1 l2 : natlist) : natlist`

[ans]

```
Fixpoint alternate (l1 l2 : natlist) : natlist :=
  match l1 with
  | nil => l2
  | h1::t1 => match l2 with
    | nil => h1::t1
    | h2::t2 => [h1;h2] ++ (alternate t1 t2)
  end
end.
```

```
Example test_alternate1:
  alternate [1;2;3] [4;5;6] = [1;4;2;5;3;6].
Proof. reflexivity. Qed.
```

```
Example test_alternate2:
  alternate [1] [4;5;6] = [1;4;5;6].
Proof. reflexivity. Qed.
```

```
Example test_alternate3:
  alternate [1;2;3] [4] = [1;4;2;3].
Proof. reflexivity. Qed.
```

```
Example test_alternate4:
  alternate [] [20;30] = [20;30].
Proof. reflexivity. Qed.
```

[运行结果]

```

Fixpoint alternate (l1 l2 : natlist) : natlist :=
  match l1 with
  | nil => l2
  | h1::t1 => match l2 with
    | nil => h1::t1
    | h2::t2 => [h1;h2] ++ (alternate t1 t2)
  end
end.

```

```

Example test_alternate1:
  alternate [1;2;3] [4;5;6] = [1;4;2;5;3;6].
Proof. reflexivity. Qed.

```

```

Example test_alternate2:
  alternate [1] [4;5;6] = [1;4;5;6].
Proof. reflexivity. Qed.

```

```

Example test_alternate3:
  alternate [1;2;3] [4] = [1;4;2;3].
Proof. reflexivity. Qed.

```

```

Example test_alternate4:
  alternate [] [20;30] = [20;30].
Proof. reflexivity. Qed.

```

2.假设我们用列表来表示集合，定义函数 **inter**使得 (**inter l1 l2**)的结果为**l1**和**l2**这两个集合的交集。

[ans]

```

Fixpoint one_exist(v:nat) (l:natlist):bool:=
  match l with
  | nil => false
  | h::t => match h=?v with
    | true => true
    | false => one_exist v t
  end
end.

Fixpoint inter (l1 l2:natlist):natlist:=
  match l1 with
  | nil => nil
  | h1::t1 => if (one_exist h1 l2) then [h1] ++ (inter t1 l2) else
(inter t1 l2)
  end.

```

```

Example test_inter1:
  inter [1;2;3] [4]=[].
Proof. reflexivity. Qed.
Example test_inter2:
  inter [1;2;3] [1;2]=[1;2].
Proof. simpl. reflexivity. Qed.
Example test_inter3:
  inter [1;2;3;4] [3;4]=[3;4].
Proof. simpl. reflexivity. Qed.
Example test_inter4:
  inter [1;2;3;4] [1;3]=[1;3].
Proof. simpl. reflexivity. Qed.
(** [] *)

```

[运行结果]

```

Fixpoint one_exist(v:nat) (l:natlist):bool:=
  match l with
  |nil=>false
  |h::t=>match h=?v with
    |true=>true
    |false=>one_exist v t
  end
end.

Fixpoint inter (l1 l2:natlist):natlist:=
  match l1 with
  |nil=>nil
  |h1::t1=>if (one_exist h1 l2) then [h1] ++ (inter t1 l2) else (inter t1 l2)
end.

Example test_inter1:
  inter [1;2;3] [4]=[].
Proof. reflexivity. Qed.
Example test_inter2:
  inter [1;2;3] [1;2]=[1;2].
Proof. simpl. reflexivity. Qed.
Example test_inter3:
  inter [1;2;3;4] [3;4]=[3;4].
Proof. simpl. reflexivity. Qed.
Example test_inter4:
  inter [1;2;3;4] [1;3]=[1;3].
Proof. simpl. reflexivity. Qed.
(** [] *)

```