```
Last login: Fri Feb  3 14:43:25 on ttys016
carbon:public-class-repo$ cd SamplePrograms/Sec_10_3\:35pm/
carbon:Sec_10_3:35pm$ utop
────────────┬────────────────────────────────────────────────────────────┬────────────
            │ Welcome to utop version 1.14 (using OCaml version 4.01.0)! │
            └────────────────────────────────────────────────────────────┘

Type #utop_help for help about using utop.

─( 18:00:00 )─< command 0 >──────────────────────────────────────{ counter: 0 }─
utop # #use "fold.ml";;
val fold : ('a -> 'b -> 'b) -> 'b -> 'a list -> 'b = <fun>
─( 15:52:54 )─< command 1 >──────────────────────────────────────{ counter: 0 }─
utop # fold (+) 0 [1;2;3;4] ;;
- : int = 10
─( 15:52:59 )─< command 2 >──────────────────────────────────────{ counter: 0 }─
utop # (^) ;;
- : string -> string -> string = <fun>
─( 15:53:15 )─< command 3 >──────────────────────────────────────{ counter: 0 }─
utop # int_of_string ;;
- : string -> int = <fun>
─( 15:56:54 )─< command 4 >──────────────────────────────────────{ counter: 0 }─
utop # string_of_int ;;
- : int -> string = <fun>
─( 15:57:06 )─< command 5 >──────────────────────────────────────{ counter: 0 }─
utop # string_of_int  4 ;;
- : string = "4"
─( 15:57:15 )─< command 6 >──────────────────────────────────────{ counter: 0 }─
utop # #use "fold.ml";;
val fold : ('a -> 'b -> 'b) -> 'b -> 'a list -> 'b = <fun>
val string_folder : int -> string -> string = <fun>
─( 15:57:19 )─< command 7 >──────────────────────────────────────{ counter: 0 }─
utop # fold string_folder "" [1;2;3;4] ;;
- : string = "4321"
─( 15:58:07 )─< command 8 >──────────────────────────────────────{ counter: 0 }─
utop # fold ;;
- : ('a -> 'b -> 'b) -> 'b -> 'a list -> 'b = <fun>
─( 15:58:17 )─< command 9 >──────────────────────────────────────{ counter: 0 }─
utop # #use "fold.ml";;
val fold : ('a -> 'b -> 'b) -> 'b -> 'a list -> 'b = <fun>
val string_folder : int -> string -> string = <fun>
─( 15:59:27 )─< command 10 >─────────────────────────────────────{ counter: 0 }─
utop # fold string_folder "" [1;2;3;4] ;;
- : string = "1 2 3 4 "
─( 15:59:42 )─< command 11 >─────────────────────────────────────{ counter: 0 }─
utop # #use "fold.ml";;
val fold_v1 : ('a -> 'b -> 'b) -> 'b -> 'a list -> 'b = <fun>
val string_folder : int -> string -> string = <fun>
val fold_v2 : ('a -> 'b -> 'a) -> 'a -> 'b list -> 'a = <fun>
─( 15:59:44 )─< command 12 >─────────────────────────────────────{ counter: 0 }─
utop # fold_v2 (+) 0 [1;2;3;4] ;;
- : int = 10
─( 16:05:12 )─< command 13 >─────────────────────────────────────{ counter: 0 }─
```

```
utop # #use "fold.ml";;
val fold_v1 : ('a -> 'b -> 'b) -> 'b -> 'a list -> 'b = <fun>
val string_folder : int -> string -> string = <fun>
val fold_v2 : ('a -> 'b -> 'b) -> 'b -> 'a list -> 'b = <fun>
─( 16:05:39 )─< command 14 >─────────────────────────────────────{ counter: 0 }─
utop # fold_v2 string_folder "" [1;2;3;4] ;;
- : string = "4 3 2 1 "
─( 16:08:42 )─< command 15 >─────────────────────────────────────{ counter: 0 }─
utop # #use "fold.ml";;
val fold_v1 : ('a -> 'b -> 'b) -> 'b -> 'a list -> 'b = <fun>
val string_folder : int -> string -> string = <fun>
val fold_v2 : ('a -> 'b -> 'a) -> 'a -> 'b list -> 'a = <fun>
─( 16:09:16 )─< command 16 >─────────────────────────────────────{ counter: 0 }─
utop # string_folder ;;
- : int -> string -> string = <fun>
─( 16:10:08 )─< command 17 >─────────────────────────────────────{ counter: 0 }─
utop # fold_v2 (-) 0 [7;4;2;] ;;
- : int = -13
─( 16:10:46 )─< command 18 >─────────────────────────────────────{ counter: 0 }─
utop # fold_v1 (-) 0 [7;4;2;] ;;
- : int = 5
─( 16:11:36 )─< command 19 >─────────────────────────────────────{ counter: 0 }─
utop # #use "fold.ml";;
val fold_v1 : ('a -> 'b -> 'b) -> 'b -> 'a list -> 'b = <fun>
val string_folder : int -> string -> string = <fun>
val fold_v2 : ('a -> 'b -> 'a) -> 'a -> 'b list -> 'a = <fun>
val foldr : ('a -> 'b -> 'b) -> 'a list -> 'b -> 'b = <fun>
val foldl : ('a -> 'b -> 'a) -> 'a -> 'b list -> 'a = <fun>
─( 16:11:48 )─< command 20 >─────────────────────────────────────{ counter: 0 }─
utop # foldl (+) 0 [1;2;3;4] ;;
- : int = 10
─( 16:15:36 )─< command 21 >─────────────────────────────────────{ counter: 0 }─
utop # foldr (+) [1;2;3;4] 0 ;;
- : int = 10
─( 16:16:02 )─< command 22 >─────────────────────────────────────{ counter: 0 }─
utop # foldl ;;
- : ('a -> 'b -> 'a) -> 'a -> 'b list -> 'a = <fun>
─( 16:16:21 )─< command 23 >─────────────────────────────────────{ counter: 0 }─
utop # #use "fold.ml";;
val fold_v1 : ('a -> 'b -> 'b) -> 'b -> 'a list -> 'b = <fun>
val string_folder : int -> string -> string = <fun>
val fold_v2 : ('a -> 'b -> 'a) -> 'a -> 'b list -> 'a = <fun>
val foldr : ('a -> 'b -> 'b) -> 'a list -> 'b -> 'b = <fun>
val foldl : ('a -> 'b -> 'a) -> 'a -> 'b list -> 'a = <fun>
val length : 'a list -> int = <fun>
─( 16:18:52 )─< command 24 >─────────────────────────────────────{ counter: 0 }─
utop # length [1;2;3;4] ;;
- : int = 4
─( 16:21:17 )─< command 25 >─────────────────────────────────────{ counter: 0 }─
utop # lengh ['a'; 'b'; 'c' ] ;;
Error: Unbound value lengh
Did you mean length?
─( 16:21:21 )─< command 26 >─────────────────────────────────────{ counter: 0 }─
```

```
utop # length ['a'; 'b'; 'c' ] ;;
- : int = 3
─( 16:21:30 )─< command 27 >─────────────────────────────────────{ counter: 0 }─
utop # #use "fold.ml";;
val fold_v1 : ('a -> 'b -> 'b) -> 'b -> 'a list -> 'b = <fun>
val string_folder : int -> string -> string = <fun>
val fold_v2 : ('a -> 'b -> 'a) -> 'a -> 'b list -> 'a = <fun>
val foldr : ('a -> 'b -> 'b) -> 'a list -> 'b -> 'b = <fun>
val foldl : ('a -> 'b -> 'a) -> 'a -> 'b list -> 'a = <fun>
val length : 'a list -> int = <fun>
val sum : int list -> int = <fun>
─( 16:21:37 )─< command 28 >─────────────────────────────────────{ counter: 0 }─
utop # sum [1;2;3;4] ;;
- : int = 10
─( 16:22:56 )─< command 29 >─────────────────────────────────────{ counter: 0 }─
utop # foldr (fun h t -> h :: t) [] (1::2::3::4::[]) ;;
- : int list = [1; 2; 3; 4]
─( 16:23:05 )─< command 30 >─────────────────────────────────────{ counter: 0 }─
utop # foldr (+) 0 (1::2::3::4::[]) ;;
Error: This expression has type int but an expression was expected of type
         int list
─( 16:26:24 )─< command 31 >─────────────────────────────────────{ counter: 0 }─
utop # foldr (+) (1::2::3::4::[]) 0;;
- : int = 10
─( 16:26:41 )─< command 32 >─────────────────────────────────────{ counter: 0 }─
utop # foldr (fun h t -> h :: t) [] (1::2::3::4::[]) ;;
- : int list = [1; 2; 3; 4]
─( 16:27:00 )─< command 33 >─────────────────────────────────────{ counter: 0 }─
utop # foldr (fun h t -> h :: t)  (1::2::3::4::[]) [] ;;
- : int list = [1; 2; 3; 4]
─( 16:27:14 )─< command 34 >─────────────────────────────────────{ counter: 0 }─
utop # foldr (+) (1::2::3::4::[]) 0;;
- : int = 10
─( 16:27:28 )─< command 35 >─────────────────────────────────────{ counter: 0 }─
utop #
┌───┬────────────┬─────┬───────────┬──────────────┬───────┬────────┬──────┬────┐
│Arg│Arith_status│Array│ArrayLabels│Assert_failure│Big_int│Bigarray│Buffer│Call│
└───┴────────────┴─────┴───────────┴──────────────┴───────┴────────┴──────┴────┘
```