```
Last login: Fri Mar 30 13:23:31 on ttys007
carbon:$ cd Search/
carbon:$ utop
                Welcome to utop version 2.0.2 (using OCaml version 4.06.0)!

Type #utop_help for help about using utop.

─( 13:31:09 )─< command 0 >──────────────────────────────────{ counter: 0 }─
utop # #use "search_options.ml";;
val gen_subsets : 'a list -> 'a list list = <fun>
val s : int list = [1; 3; -2; 5; -6]
val sum : int list -> int = <fun>
val subsetsum_option_v1 : 'a list -> 'a list option = <fun>
val subsetsum_option_v2 : int list -> int list = <fun>
val show_list : ('a -> string) -> 'a list -> string = <fun>
val process_solution_option : ('a -> string) -> 'a -> 'a option = <fun>
val subsetsum_option : int list -> int list option = <fun>
─( 13:31:09 )─< command 1 >──────────────────────────────────{ counter: 0 }─
utop # gen_subsets [1;2;3] ;;
- : int list list = [[]; [1]; [2]; [2; 1]; [3]; [3; 1]; [3; 2]; [3; 2; 1]]
─( 13:31:13 )─< command 2 >──────────────────────────────────{ counter: 0 }─
utop # #use "search_options.ml";;
val gen_subsets : 'a list -> 'a list list = <fun>
val gen_subsets_v2 : 'a list -> 'a list list = <fun>
val s : int list = [1; 3; -2; 5; -6]
val sum : int list -> int = <fun>
val subsetsum_option_v1 : 'a list -> 'a list option = <fun>
val subsetsum_option_v2 : int list -> int list = <fun>
val show_list : ('a -> string) -> 'a list -> string = <fun>
val process_solution_option : ('a -> string) -> 'a -> 'a option = <fun>
val subsetsum_option : int list -> int list option = <fun>
─( 13:31:23 )─< command 3 >──────────────────────────────────{ counter: 0 }─
utop # gen_subsets_v2 [1;2;3] ;;
- : int list list = [[3; 2; 1]; [2; 1]; [3; 1]; [1]; [3; 2]; [2]; [3]; []]
─( 13:35:47 )─< command 4 >──────────────────────────────────{ counter: 0 }─
utop # #use "search_options.ml";;
val gen_subsets : 'a list -> 'a list list = <fun>
val gen_subsets_v2 : 'a list -> 'a list list = <fun>
val s : int list = [1; 3; -2; 5; -6]
val sum : int list -> int = <fun>
val subsetsum_option_v1 : int list -> int list option = <fun>
val subsetsum_option_v2 : int list -> int list = <fun>
val show_list : ('a -> string) -> 'a list -> string = <fun>
val process_solution_option : ('a -> string) -> 'a -> 'a option = <fun>
val subsetsum_option : int list -> int list option = <fun>
─( 13:35:52 )─< command 5 >──────────────────────────────────{ counter: 0 }─
utop # subsetsum_option_v1 s ;;
- : int list option = Some [-6; 5; 1]
─( 13:43:27 )─< command 6 >──────────────────────────────────{ counter: 0 }─
```

```
utop # #use "search_options.ml";;
val gen_subsets : 'a list -> 'a list list = <fun>
val gen_subsets_v2 : 'a list -> 'a list list = <fun>
val s : int list = [1; 3; -2; 5; -6]
val sum : int list -> int = <fun>
val subsetsum_option_v1 : int list -> int list option = <fun>
val subsetsum_option_v2 : int list -> int list = <fun>
val show_list : ('a -> string) -> 'a list -> string = <fun>
val process_solution_option : ('a -> string) -> 'a -> 'a option = <fun>
val subsetsum_option : int list -> int list option = <fun>
─( 13:43:46 )─< command 7 >──────────────────────────────────────{ counter: 0 }─
utop # subsetsum_option_v1 s ;;
- : int list option = Some [-6; 5; 1]
─( 13:45:18 )─< command 8 >──────────────────────────────────────{ counter: 0 }─
utop # #use "search_options.ml";;
val gen_subsets : 'a list -> 'a list list = <fun>
val gen_subsets_v2 : 'a list -> 'a list list = <fun>
val s : int list = [1; 3; -2; 5; -6]
val sum : int list -> int = <fun>
val subsetsum_option_v1 : int list -> int list option = <fun>
val subsetsum_option_v2 : int list -> int list = <fun>
val show_list : ('a -> string) -> 'a list -> string = <fun>
val process_solution_option : ('a -> string) -> 'a -> 'a option = <fun>
val subsetsum_option : int list -> int list option = <fun>
─( 13:45:19 )─< command 9 >──────────────────────────────────────{ counter: 0 }─
utop # subsetsum_option_v1 s ;;
- : int list option = Some [-6; 5; 1]
─( 13:45:22 )─< command 10 >─────────────────────────────────────{ counter: 0 }─
utop # #use "search_options.ml";;
val gen_subsets : 'a list -> 'a list list = <fun>
val gen_subsets_v2 : 'a list -> 'a list list = <fun>
val s : int list = [1; 3; -2; 5; -6]
val sum : int list -> int = <fun>
File "search_options.ml", line 67, characters 16-31:
Error: This expression has type int list option
       but an expression was expected of type 'a list
─( 13:45:24 )─< command 11 >─────────────────────────────────────{ counter: 0 }─
utop # #use "search_options.ml";;
val gen_subsets : 'a list -> 'a list list = <fun>
val gen_subsets_v2 : 'a list -> 'a list list = <fun>
val s : int list = [1; 3; -2; 5; -6]
val sum : int list -> int = <fun>
val subsetsum_option_v1 : int list -> int list option = <fun>
val subsetsum_option_v2 : int list -> int list = <fun>
val show_list : ('a -> string) -> 'a list -> string = <fun>
val process_solution_option : ('a -> string) -> 'a -> 'a option = <fun>
val subsetsum_option : int list -> int list option = <fun>
─( 13:46:10 )─< command 12 >─────────────────────────────────────{ counter: 0 }─
utop # subsetsum_option_v2 s ;;
- : int list = [-6; 5; 1]
─( 13:49:26 )─< command 13 >─────────────────────────────────────{ counter: 0 }─
```

```
utop # #use "search_options.ml";;
val gen_subsets : 'a list -> 'a list list = <fun>
val gen_subsets_v2 : 'a list -> 'a list list = <fun>
val s : int list = [1; 3; -2; 5; -6]
val sum : int list -> int = <fun>
val subsetsum_option_v1 : int list -> int list option = <fun>
val subsetsum_option_v2 : int list -> int list = <fun>
val show_list : ('a -> string) -> 'a list -> string = <fun>
val process_solution_option : ('a -> string) -> 'a -> 'a option = <fun>
File "search_options.ml", line 147, characters 37-41:
Error: Unbound value show
-( 13:49:31 )-< command 14 >-------------------------------------{ counter: 0 }-
utop # #use "search_options.ml";;
val gen_subsets : 'a list -> 'a list list = <fun>
val gen_subsets_v2 : 'a list -> 'a list list = <fun>
val s : int list = [1; 3; -2; 5; -6]
val sum : int list -> int = <fun>
val subsetsum_option_v1 : int list -> int list option = <fun>
val subsetsum_option_v2 : int list -> int list = <fun>
val show_list : ('a -> string) -> 'a list -> string = <fun>
val process_solution_option : ('a -> string) -> 'a -> 'a option = <fun>
File "search_options.ml", line 147, characters 37-46:
Error: This expression has type ('a -> string) -> 'a list -> string
       but an expression was expected of type ('a -> string) -> string
       Type 'a list -> string is not compatible with type string
-( 13:55:03 )-< command 15 >-------------------------------------{ counter: 0 }-
utop # #use "search_options.ml";;
val gen_subsets : 'a list -> 'a list list = <fun>
val gen_subsets_v2 : 'a list -> 'a list list = <fun>
val s : int list = [1; 3; -2; 5; -6]
val sum : int list -> int = <fun>
val subsetsum_option_v1 : int list -> int list option = <fun>
val subsetsum_option_v2 : int list -> int list = <fun>
val show_list : ('a -> string) -> 'a list -> string = <fun>
val process_solution_option : ('a -> string) -> 'a -> 'a option = <fun>
val subsetsum_option : int list -> int list option = <fun>
-( 13:55:17 )-< command 16 >-------------------------------------{ counter: 0 }-
utop # subsetsum_option s ;;
Here is a solution: [ -6; 5; 1 ]
Do you like it ?
n
Here is a solution: [ -6; 5; -2; 3 ]
Do you like it ?
n
- : int list option = None
-( 13:56:27 )-< command 17 >-------------------------------------{ counter: 0 }-
utop # subsetsum_option s ;;
Here is a solution: [ -6; 5; 1 ]
Do you like it ?
Y
Thanks for playing...
```

```
- : int list option = Some [-6; 5; 1]
-( 13:57:00 )-< command 18 >-------------------------------------{ counter: 0 }-
utop #
```

Arg	Array	ArrayLabels	Assert_failure	Bigarray	Buffer	Bytes	BytesLabels	Call