```
Last login: Mon Jan 23 15:21:53 on ttys022
carbon:SamplePrograms$ 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 # let inc x = x + 1 ;;
val inc : int -> int = <fun>
─( 15:32:06 )─< command 1 >──────────────────────────────────────{ counter: 0 }─
utop # inc 4 ;;
- : int = 5
─( 15:37:16 )─< command 2 >──────────────────────────────────────{ counter: 0 }─
utop # let add x y = x + y ;;
val add : int -> int -> int = <fun>
─( 15:37:18 )─< command 3 >──────────────────────────────────────{ counter: 0 }─
utop # let add' = fun x -> (fun y -> x + y) ;;
val add' : int -> int -> int = <fun>
─( 15:37:23 )─< command 4 >──────────────────────────────────────{ counter: 0 }─
utop # let power x y = if x = 0 then y else y * power x - 1 ;;
Error: Unbound value power
─( 15:38:22 )─< command 5 >──────────────────────────────────────{ counter: 0 }─
utop # let power x y = if x = 0 then y else power x-1 y *. y ;;
Error: Unbound value power
─( 15:47:29 )─< command 6 >──────────────────────────────────────{ counter: 0 }─
utop # let rec power x y = if x = 0 then y else power x-1 y *. y ;;
Error: This expression has type 'a -> 'a but an expression was expected of type
         int
─( 15:48:12 )─< command 7 >──────────────────────────────────────{ counter: 0 }─
utop # let rec power x y = if x = 0 then y else power (x-1) y *. y ;;
val power : int -> float -> float = <fun>
─( 15:48:25 )─< command 8 >──────────────────────────────────────{ counter: 0 }─
utop # power 3 3.0 ;;
- : float = 81.
─( 15:49:47 )─< command 9 >──────────────────────────────────────{ counter: 0 }─
utop # let rec power x y = if x = 0 then 1.0 else power (x-1) y *. y ;;
val power : int -> float -> float = <fun>
─( 15:50:03 )─< command 10 >─────────────────────────────────────{ counter: 0 }─
utop # power 3 3.0 ;;
- : float = 27.
─( 15:50:24 )─< command 11 >─────────────────────────────────────{ counter: 0 }─
utop # power 3 3.2 ;;
- : float = 32.768000000000078
─( 15:50:25 )─< command 12 >─────────────────────────────────────{ counter: 0 }─
utop # let cube x = power 3 x ;;
val cube : float -> float = <fun>
─( 15:52:13 )─< command 13 >─────────────────────────────────────{ counter: 0 }─
utop # cube 3.0 ;
;;
- : float = 27.
─( 15:54:35 )─< command 14 >─────────────────────────────────────{ counter: 0 }─
```

```
utop # let cube = power 3 ;;
val cube : float -> float = <fun>
─( 15:54:39 )─< command 15 >──────────────────────────────────{ counter: 0 }─
utop # cube 3.0 ;
;;
- : float = 27.
─( 15:55:42 )─< command 16 >──────────────────────────────────{ counter: 0 }─
utop # power ;;
- : int -> float -> float = <fun>
─( 15:55:46 )─< command 17 >──────────────────────────────────{ counter: 0 }─
utop # let square = power 2 ;;
val square : float -> float = <fun>
─( 15:56:01 )─< command 18 >──────────────────────────────────{ counter: 0 }─
utop # square 4 ;;
Error: This expression has type int but an expression was expected of type
         float
─( 15:56:53 )─< command 19 >──────────────────────────────────{ counter: 0 }─
utop # square 4.0 ;;
- : float = 16.
─( 15:56:56 )─< command 20 >──────────────────────────────────{ counter: 0 }─
utop # let x
     = 3 + 6
     * 7 ;;
val x : int = 45
─( 15:57:01 )─< command 21 >──────────────────────────────────{ counter: 0 }─
utop # and ;;
Error: Syntax error
─( 16:00:17 )─< command 22 >──────────────────────────────────{ counter: 0 }─
utop # (&&) ;;
- : bool -> bool -> bool = <fun>
─( 16:12:31 )─< command 23 >──────────────────────────────────{ counter: 0 }─
utop # true && false ;;
- : bool = false
─( 16:12:40 )─< command 24 >──────────────────────────────────{ counter: 0 }─
utop # #use "gcd.ml";;
val gcd : int -> int -> int = <fun>
─( 16:12:50 )─< command 25 >──────────────────────────────────{ counter: 0 }─
utop # gcd 4 10 ;;
- : int = 2
─( 16:13:32 )─< command 26 >──────────────────────────────────{ counter: 0 }─
utop # gcd 1 10
;;
- : int = 1
─( 16:13:39 )─< command 27 >──────────────────────────────────{ counter: 0 }─
utop #
```