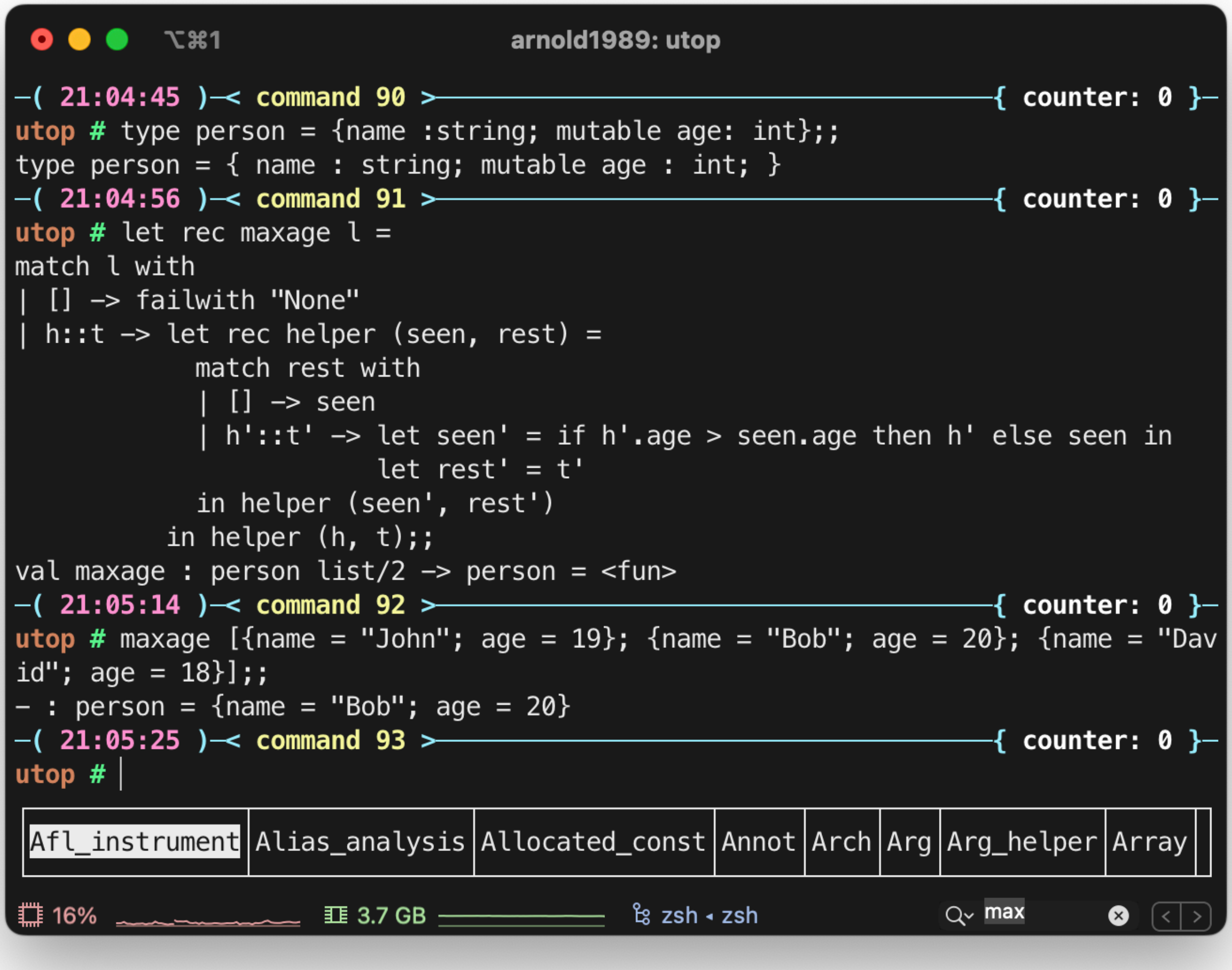

```
─( 21:04:45 )─< command 90 >──────────────────────────────────────────{ counter: 0 }─
utop # type person = {name :string; mutable age: int};;
type person = { name : string; mutable age : int; }
─( 21:04:56 )─< command 91 >──────────────────────────────────────────{ counter: 0 }─
utop # let rec maxage l =
match l with
| [] -> failwith "None"
| h::t -> let rec helper (seen, rest) =
            match rest with
            | [] -> seen
            | h'::t' -> let seen' = if h'.age > seen.age then h' else seen in
                        let rest' = t'
            in helper (seen', rest')
          in helper (h, t);;
val maxage : person list/2 -> person = <fun>
─( 21:05:14 )─< command 92 >──────────────────────────────────────────{ counter: 0 }─
utop # maxage [{name = "John"; age = 19}; {name = "Bob"; age = 20}; {name = "Dav
id"; age = 18}];;
- : person = {name = "Bob"; age = 20}
─( 21:05:25 )─< command 93 >──────────────────────────────────────────{ counter: 0 }─
utop # |
┌──────────────┬──────────────┬───────────────┬─────┬────┬───┬──────────┬─────┬┐
│Afl_instrument│Alias_analysis│Allocated_const│Annot│Arch│Arg│Arg_helper│Array││
└──────────────┴──────────────┴───────────────┴─────┴────┴───┴──────────┴─────┴┘
```

没什么思维难度，按照课件中给的例子来就可以了。语法和coq差不多，结构体的形式与C差不多。