Write the *type* of each of the following OCaml expressions in the first blank provided, or *ill-typed* if the expression does not type check. Then, after the ⇒ symbol, write the most simplified *value* of the expression, or leave it blank if it's ill-typed. We have done the first one for you as an example.

(a) `let a :` _____*int*_____ `= 3 + 4` ⇒ _____7_____

(b) `let b :` _____ `= a = 8` ⇒ _____

(c) `let c :` _____ `= 5 / 4` ⇒ _____

(d) `let d :` _____ `= "hello " + "world"` ⇒ _____

(e) `let e :` _____ `= if a > 0 then "positive"` ⇒ _____

(f) `let f :` _____ `= if a > 0 then "positive"`
                      `else "negative"` ⇒ _____

(g) `let g :` _____ `= if a > 0 then 42 else 41.5` ⇒ _____

(h) `let h :` _____ `= let q = 3 <> 4 in`
                      `(not q) && (a = 7)` ⇒ _____

Locate each binding in the code below. Then determine the scope of each identifier. (Give the range of lines.) If the binding is shadowed by a later one, indicate that as well. The first two are done as an example.

```
1          let profit_500 : int =
2                  let price    = 500 in
3                  let attendees = 120 in
4                  let revenue  = price * attendees in
5                  let cost     = 18000 + 4 * attendees in
6                  revenue - cost
7
8          let attendees (ticket_price : int) : int =
9             (-15 * ticket_price)/10 + 870
10
11         let test () : bool =
12            (attendees 500) = 120
13         ;; run_test "attendees at $5.00" test
14
15         let test () : bool =
16            (attendees 480) = 150
17         ;; run_test "attendees at $4.80" test
18
19         let cost (ticket_price : int) : int =
20            18000 + (attendees ticket_price) * 4
21
22         let revenue (ticket_price : int) =
23            (attendees ticket_price)  * ticket_price
24
25         let profit (ticket_price : int) : int =
26            (revenue ticket_price) - (cost ticket_price)
27
28         let test () : bool =
29            (profit 500) = profit_500
30         ;; run_test "profit at $5.00" test
```

Bound: profit_500, lines 7-30
Bound: price, lines 3-6