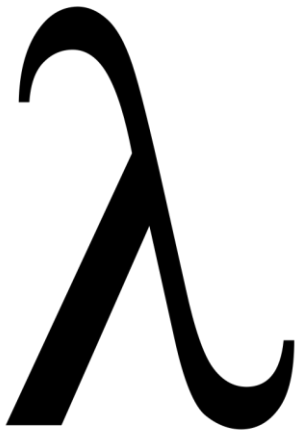

ML Lab 1

Programmazione Funzionale

2024/2025

Università di Trento

Chiara Di Francescomarino

Some other questions

How to participate?

Exercise 1.1

- Write the expression to convert
  - 123.45 to the next lower integer
  - -123.45 to the next lower integer
  - 123.45 to the next higher integer
  - -123.45 to the next higher integer

Solutions 1.1

```
> floor 123.45;
val it = 123: int
> floor ~123.45;
val it = ~124: int
> ceil 123.45;
val it = 124: int
> ceil ~123.45;
val it = ~123: int
```

Exercise 1.2

- Write the expression to convert
  - #”Y” to an integer
  - 120 to a character
  - 97.0 to a character
  - #”N” to a real
  - #”Z” to a string

Solutions 1.2

```
> ord #"Y";
val it = 89: int
> chr 120;
val it = #"x": char
> chr(round(97.0));
val it = #"a": char
> real(ord(#"N"));
val it = 78.0: real
> str #"Z";
val it = "Z": string
```

Exercise 1.3

- How can we fix the errors in the following expressions?
  - `ceil(4);`
  - `if true then 5+6 else 7.0;`

Solutions 1.3

```
> ceil(4);
poly: : error: Type error in function application.
   Function: ceil : real -> int
Static Errors
> ceil (4.0);
val it = 4: int

> if true then 5+6 else 7.0;
poly: : error: Type mismatch between then-part and else-part.
   Then: 5 + 6 : int
   Else: 7.0 : real
> if true then 5+6 else 7;
val it = 11: int
```

Exercise 1.4

- How can we fix the errors in the following expressions?
  - `if 0 then 1 else 2;`
  - `ord("a")`

Solutions 1.4

```
> if 0 then 1 else 2;
poly: : error: Condition in
if-statement must have type
bool.
   If: 0 : int
   Reason:
      Can't unify int (*In
Basis*) with bool (*In Basis*)
         (Different type
constructors)
Found near if 0 then 1 else 2
Static Errors
> if false then 1 else 2;
val it = 2: int
```

```
> ord("a");
poly: : error: Type error in
function application.
   Function: ord : char -> int
   Argument: ("a") : string
   Reason:
      Can't unify char (*In
Basis*) with string (*In
Basis*)
         (Different type
constructors)
Found near ord ("a")
Static Errors
> ord(#"a");
val it = 97: int
```

Few more questions

How to participate?

Copy participation link

1 Go to wooclap.com

2 Enter the event code in the top banner

Event code
RTJGCD

Exercise 1.5

- Fix the following expressions

```
explode ["bar"];

implode ( #"a", #"b") ;

["r"]::["a","t"];
```

Solutions 1.5

```
> explode ["bar"];
poly: : error: Type error in function application.
Function: explode : string -> char list
Argument: ["bar"] : string list
> explode "bar";
val it = [#"b", #"a", #"r"]: char list

> implode ( #"a", #"b") ;
poly: : error: Type error in function application.
Function: implode : char list -> string
implode [ #"a", #"b"] ;
> implode [ #"a", #"b"] ;
val it = "ab": string

> ["r"]::["a","t"];
poly: : error: Type error in function application.
Function: :: : string list * string list list -> string list list
Argument: (["r"], ["a", "t"]) : string list * string list
> "r"::["a","t"];
val it = ["r", "a", "t"]: string list
> ["r"]@["a", "t"];
val it = ["r", "a", "t"]: string list
```

Exercise 1.6

- Give examples of objects of the following types, without using empty lists

```
int list list list

(int * char) list

string list * ( int * (real * string))
* int
```

Solutions 1.6

```
> [[[1,2]],[[3,4]]];
val it = [[[1, 2]], [[3, 4]]]: int list list
list

> [(1,#"a"), (2,#"b")];
val it = [(1, #"a"), (2, #"b")]: (int * char)
list

> ( ["ab","cd"], (4,(2.5,"ef")), 7);
val it = (["ab", "cd"], (4, (2.5, "ef")), 7):
string list * (int * (real * string)) * int
```

Exercise 1.7

- Give examples of objects of the following types, without using empty lists

```
((int * int) * (bool list) * real) *
(real * string)

(bool * int) * char

real * int list list list list
```

Solutions 1.7

```
> (((5,6), [true, false], 5.6), (6.7, "abc"));
val it = (((5, 6), [true, false], 5.6), (6.7,
"abc")):
((int * int) * bool list * real) * (real *
string)


> ((true, 7), #"a");
val it = ((true, 7), #"a"): (bool * int) * char


> (7.8, [[[[1,2],[3,4]]]]);
val it = (7.8, [[[[1, 2], [3, 4]]]]): real * int
list list list list
```

Exercise L1.8

- Write a function `cube` that computes the cube of a real number
- For instance
  - `cube (2.9) = 24.389`

Solutions L1.8

```
> fun cube (x:real) = x * x * x;
val cube = fn: real -> real

> cube (2.9);
val it = 24.389: real

> cube 2;
poly: : error: Type error in function
application.
Function: cube : real -> real
Argument: 2 : int
```

Exercise L1.9

- Write a function `min3` that computes the smallest component of a tuple of type int*int*int
- For instance
  - `min3 (2,3,4) = 2`
  - `min3 (3,2,4) = 2`

Solution L1.9

```
> fun min3 (a,b,c) =
  if a<b then
        if a<c then a else c
  else
        if b<c then b else c;
val min3 = fn: int * int * int -> int

> min3 (2,3,4);
val it = 2: int
> min3 (3,2,4);
val it = 2: int
> min3 (4,3,2);
val it = 2: int
```