

Tipi per il progetto

Di seguito elenco i tipi corretti. Nel documento che descrive il progetto ci sono alcune imprecisioni.

```
type ide = Ide of string
```

Il tipo **exp** corretto è il seguente. La differenza sta nel costruttore per le coppie che deve essere distinto da quello dei valori restituiti, quindi qui si chiama **Epair** e non **Pair**

```
type exp =  
  Val of ide  
| Eint of int  
| Echar of char  
| True  
| False  
| Empty  
| Sum of exp * exp  
| Diff of exp * exp  
| Times of exp * exp  
| And of exp * exp  
| Or of exp * exp  
| Not of exp  
| Eq of exp * exp  
| Less of exp * exp  
| Cons of exp * exp  
| Head of exp  
| Tail of exp  
| Fst of exp  
| Snd of exp  
| Epair of exp * exp  
| Ifthenelse of exp * exp * exp  
| Let of ide * exp * exp  
| Fun of ide * exp  
| Appl of exp * exp  
| Rec of ide * exp;
```

```
type eval =
    Undefined
  | Int of int
  | Bool of bool
  | Char of char
  | List of eval list
  | Pair of eval * eval
  | Closure of exp * env;;
```

Nel testo del progetto ho indicato questo tipo per l'ambiente:

```
type env = string -> eval;;
```

In effetti è meglio usare questo:

```
type env = ide -> eval;;
```

anche perchè le funzioni sull'ambiente (`emptyenv`, `applyenv`, `bind`) prendono identificatori

In questo tipo mancava un caso (caratteri). Ora c'è, ed è stato corretto anche lo spelling.

```
type etype =
    TBool
  | TInt
  | TChar
  | TVar of string
  | TPair of etype * etype
  | TList of etype list
  | TFun of etype * etype;;
```