

**Prova Scritta di Linguaggi di programmazione**  
**Corso di Linguaggi di programmazione II**  
02/12/2004

## Prolog

Scrivere in SICSTUS PROLOG un predicato `mins2/3` che presa una lista di naturali contenente almeno due elementi dispari restituisce i due elementi dispari più piccoli presenti nella lista. Quindi, per esempio, il goal `mins2([4,2,5,3],5,3)` è vero.

Si scriva il programma con variabili anonime ove possibile.

### *Versione "dichiarativa"*

```
mins2(Xs,M1,M2) :- filterodd(Xs,Ys), minimals(Ys,M1,M2).  
  
filterodd([],[]).  
filterodd([X|Xs],Ys) :- 0 is X mod 2, filterodd(Xs,Ys).  
filterodd([X|Xs],[X|Ys]) :- 1 is X mod 2, filterodd(Xs,Ys).  
  
minimals([X,Y],X,Y) :- X <= Y.  
minimals([X,Y],Y,X) :- X > Y.  
minimals([Z|Xs],Z,X) :- minimals(Xs,X,_), Z < X.  
minimals([Z|Xs],X,Z) :- minimals(Xs,X,Y), Z >= X, Z < Y.  
minimals([Z|Xs],X,Y) :- minimals(Xs,X,Y), Z >= Y.
```

### *Versone "tosta"*

```
mins2([X|Xs],M1,M2) :- 0 is X mod 2, mins2(Xs,M1,M2).  
mins2([X|Xs],M1,M2) :- 1 is X mod 2, mins2(X,Xs,M1,M2).  
  
mins2(Y,[X|Xs],M1,M2) :- 0 is X mod 2, mins2(Y,Xs,M1,M2).  
mins2(Y,[X|Xs],M1,M2) :- 1 is X mod 2, X <= Y,  
mins2(X,Y,Xs,M1,M2).  
mins2(Y,[X|Xs],M1,M2) :- 1 is X mod 2, X > Y,  
mins2(Y,X,Xs,M1,M2).  
  
mins2(Y,Z,[],Y,Z).  
mins2(Y,Z,[X|Xs],M1,M2) :- 0 is X mod 2, mins2(Y,Z,Xs,M1,M2).  
mins2(Y,Z,[X|Xs],M1,M2) :-  
    1 is X mod 2,  
    updatemins(Y,Z,X,Y1,Z1),  
    mins2(Y1,Z1,Xs,M1,M2).  
  
updatemins(X,_,Z,Z,X) :- Z < X.  
updatemins(X,Y,Z,X,Z) :- Z >= X, Z < Y.  
updatemins(X,Y,Z,X,Y) :- Z >= Y.
```