```
<?xml version="1.0" ?><!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head><title>Logaritmo in algebra modulare</title></head><body>
<br/><hh2>Logaritmo modulare</hh2><pre><ii>
Scelgo un primo P e calcolo la sua radice primitiva G:</ii>
```

```
(%i1) P:2^16+1$  print("Uso il primo ",P)$
      print("Verifico che sia un numero primo: ",primep(P))$
      G:zn_primroot(P)$
      print("La piu' piccola radice primitiva di ",P,
      " e': ",G)$
Uso il primo 65537
Verifico che sia un numero primo: true
La piu' piccola radice primitiva di 65537 e': 3
```

```
<ii>Ora calcolo l'esponente da usare per ottenere Z
usando come base G ed usando come modulo P.
Il numero P deve essere un numero primo ossia scrivendo
<bb>primep(P)</bb> si deve ottenere true
La base G deve essere una "radice primitiva" ossia un
numero che elevato ad un qualsiasi intero tra
0 ed P-1 deve dare sempre un valore diverso tra 1 ed P-1.
Per conoscere il piu' piccolo intero positivo che goda
della proprieta' di essere una radice primitiva si puo'
usare la funzione <bb>zn_primroot(P)</bb>.
Il logaritmo discreto si calcola con questa funzione:
<bb>L:zn_log(Z,G,P)</bb>;
Usando numeri reali si tratterebbe di calcolare il logaritmo
di Z in base G ma qui si ottiene un intero modulo il numero
primo P. Ad esempio:</ii>
```

```
(%i6) Z:2$ L:zn_log(Z,G,P)$ print("Il logaritmo modulare di ",
      Z," in base ",G," modulo ",P," e' ",L)$
Il logaritmo modulare di 2 in base 3 modulo 65537 e' 55296
```

```
<ii>La potenza di G, elevata alla L usando come modulo P si
calcola in questo modo:      <bb>Z:power_mod(G,L,P)</bb>;
Verifica che il logaritmo modulare e' giusto:</ii>
```

```
(%i9) print("Verifica: ",power_mod(G,L,P))$
Verifica: 2
```

```
</pre></body><style type="text/css">
pre {color:blue}
hh2 {color:navy; background:yellow; font-size:150%}
bb {color:red }
ii {background:#bbffbb; }
</style></html>
```