```
\\ create the list of possible baskets for Q-Fano 3-folds
\\ the result is printed to the text file list_baskets.out
\\ in the following format:
\\ [BASKET, -K\cdot c_2, GGI,LENGTH]
\\GGI -- GLOBAL_GORENSTEIN_INDEX

LENGTH_MAX=16;
\\ maximal number of singular points in the basket (should be 16)
INDEX_MIN=2; \\ should be 2
INDEX_MAXX=24; \\ should be 24
\\ maximal and minimal indices of singularities

extern("rm list_baskets.out");
{
nomer=0;
for (LENGTH=1, LENGTH_MAX,
if (LENGTH>1, INDEX_MAX=min(INDEX_MAXX, 24-3*(LENGTH-1)/2 +1 ););
if (LENGTH==1, INDEX_MAX=INDEX_MAXX;);

INTERVALS_BASKET=vector(LENGTH,i,[INDEX_MIN,INDEX_MAX]);
forvec(INDEX=INTERVALS_BASKET,

Kc=24;
for (i=1,LENGTH, Kc=Kc-(INDEX[i]-1/INDEX[i]););
if (Kc>=0,
INTERVALS_WEIGHT=vector(LENGTH, i, [1,max(INDEX[i]/2,1)]);
forvec (WEIGHT=INTERVALS_WEIGHT,
TEST=1;
for (i=1,LENGTH, TEST=TEST && (gcd(WEIGHT[i],INDEX[i])==1););
for (i=1,LENGTH-1, if(INDEX[i]==INDEX[i+1], TEST=TEST && WEIGHT[i]<=WEIGHT[i+1]; ););
if (TEST,
BASKET=vector(LENGTH,i,[INDEX[i],WEIGHT[i]]);
GGI=1;
for (i=1,LENGTH,  GGI=lcm(GGI,INDEX[i]); );
write("list_baskets.out", [BASKET, Kc, GGI, LENGTH]);

\\ optional
nomer=nomer+1;
print("write to list_baskets.out Q-Fano   No. ", nomer);
\\
);
,flag=0);
);
,flag=1);
);
NN=matsize(readvec("list_baskets.out"))[2];
print("number of Q-Fanos=  ",NN);
}
```