#define ll long long  
#define N 1000020  
  
vector<int> prime ;  
vector<int> factors ;  
bool Isprime[N] ;  
void primeSieve( )  
{  
 memset(Isprime,true,sizeof(Isprime)) ;  
  
 Isprime[0] = Isprime[1] = false ;  
 Isprime[2] = true ;  
  
 for(int i=4 ; i<=N ; i+=2 ) { Isprime[i]= false ; } /\*Remove multiples of 2 \*/  
  
 int sqrtN = sqrt(N) ;  
  
 for(int i=3 ; i<= sqrtN ; i+= 2 )  
 {  
 if(Isprime[i] == true )  
 {  
 for(int j= i\*i ; j<=N ; j+= 2 \* i )  
 {  
 Isprime[j] = false ;  
 }  
 }  
 }  
  
 prime.push\_back(2) ;  
 for(int i=3 ; i<N ; i+=2 )  
 {  
 if(Isprime[i] == true )  
 {  
 prime.push\_back(i) ;  
 }  
  
 }  
  
  
}  
  
// code of prime factorization  
void factorize(int n)  
{  
 int sqrtn = sqrt(n) ;  
  
 for(int i=0 ; i<prime.size() && prime[i]<=sqrtn ; i++)  
 {  
 if( n%prime[i] == 0)  
 {  
 while( n%prime[i] == 0) // check how many times it divides n by prime[i]  
 {  
 n/=prime[i] ;  
 factors.push\_back(prime[i]);  
 }  
 }  
 sqrtn = sqrt(n) ;  
 }  
  
 if(n != 1)  
 {  
 factors.push\_back(n) ;  
 }  
}