//Miller Robin

#include<bits/stdc++.h>

#define ll long long int

#define mod 1000000007

#define MAX 10000007

using namespace std;

ll mulmod(ll a , ll b , ll mo)

{

ll q = ((long double) a \* (long double) b / (long double) mo);

ll res = a \* b - mo \* q;

return ((res % mo) + mo) % mo;

}

/\*

ll mulmod(ll a,ll b,ll c)

{

///this function calculates (a\*b)%c taking into account that a\*b might overflow

ll x = 0,y=a%c;

while(b > 0)

{

if(b%2 == 1)

{

x = (x+y)%c;

}

y = (y\*2)%c;

b /= 2;

}

return x%c;

}

\*/

ll bigmod (ll a, ll b, ll c)

{

ll res = 1;

a=a%c;

while (b > 0)

{

if (b % 2 == 1)

{

res=mulmod(res,a,c);

}

a=mulmod(a,a,c);

b=b/2;

}

return res;

}

bool miller(ll a, ll d, ll p)

{

ll x = bigmod(a,d,p);

if(x == 1 || x == p - 1)

return true;

while(d != p - 1)

{

x=mulmod(x,x,p);

d \*= 2;

if(x == 1)

{

return false;

}

if(x == p - 1)

{

return true;

}

}

return false;

}

bool isPrimes(ll p)

{

if(p<2)

{

return false;

}

if(p==2)

return true;

if(p!=2 && p%2==0)

{

return false;

}

ll d=p-1;

while(d%2==0)

d=d/2;

for(ll i=1; i<20; i++)

{

ll a=abs(rand()%(p-2))+2;

if(!miller(a,d,p))

return false;

}

return true;

}

int main()

{

ll t,n,q,i,j,ans,people,y,x,f,k;

scanf("%lld",&t);

while(t--)

{

scanf("%lld",&n);

for(i=n-1;; i--)

{

if(isPrimes(i))

{

printf("%lld\n",i);

break;

}

}

}

}