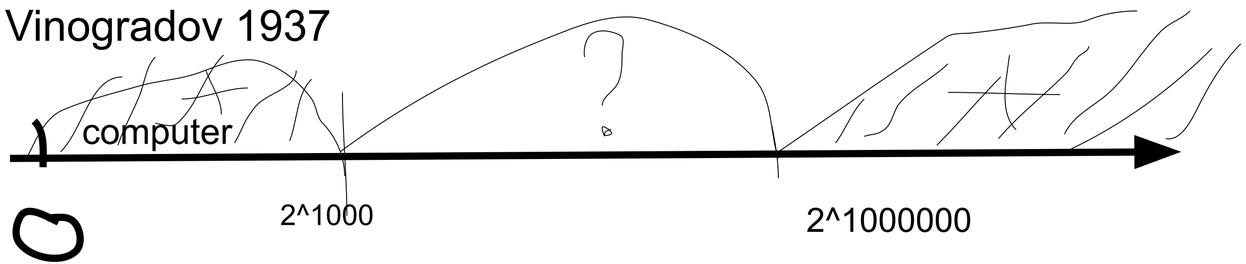
6=3+3
8=3+5
10=5+5=7+3
12=5+7
14=7+7
16=11+5
18=13+5
20=13+7
22=11+11
24=11+13
26=13+13
28=11+17=5+23
30=11+19=23+7
32 = 13 + 19
34 = 17 + 17
36 = 17 + 19
38 = 19 + 19
40 = 23 + 17




```
def goldbax(evennumber):
    ...
    if test_na_prosototu2(i)==1:
        ...
    print(prime1)
    print(prime2)
```

любое четное число можно представить в виде суммы 2-х простых

Goldbach's conjecture

350 years

```
def test_na_prosototu(number):
    flag=0
    i=2
    root=number**(0.5)
    while i<=root:
        if number%i==0:
            flag=1
            break
        i+=1
    if flag==1:
        print("not prime")
    if flag==0:
        print("prime")

test_na_prosototu(25)
```

```
def test_na_prosototu2(number):
    flag=0
    i=2
    root=number**(0.5)
    while i<=root:
        if number%i==0:
            flag=1
            break
        i+=1
    if flag==1:
        return 0//not prime
    if flag==0:
        return 1//prime

result=test_na_prosototu2(25)
if result==1:#если простое
    …
else:#если не простое
    ...
```

```
def test_na_prosototu2(number):
    flag=0
    i=2
    root=number**(0.5)
    while i<=root:
        if number%i==0:
            flag=1
            break
        i+=1
    if flag==1:
        return 0#not prime
    if flag==0:
        return 1#prime

def goldbax(evennumber):
    sposob=0
    j=2
    number=evennumber//2
    while j<=number:
        sposob=evennumber-j
        if test_na_prosototu2(sposob)==1 and test_na_prosototu2(j)==1:
            print(sposob,'+',j)
        j+=1

goldbax(1500)
```