Euler's project problem 8

John Fox

January 24, 2023

Problem statement

The four adjacent digits in the 1000-digit number that have the greatest product are $9 \times 9 \times 8 \times 9 = 5832$.
7316717653133062491922511967442657474235534919493496983520312774506326239578318016984801869478851843
8586156078911294949545950173795833195285320880551112540698747158523863050715693290963295227443043557
6689664895044524452316173185640309871112172238311362229893423380308135336276614282806444486645238749
3035890729629049156044077239071381051585930796086670172427121883998797908792274921901699720888093776
6572733300105336788122023542180975125454059475224352584907711670556013604839586446706324415722155397
5369781797784617406495514929086256932197846862248283972241375657056057490261407972968652414535100474
8216637048440319989000889524345065854122758866688116427171479924442928230863465674813919123162824586
1786645835912456652947654568284891288314260769004224219022671055626321111109370544217506941658960408
0719840385096245544436298123098787992724428490918884580156166097919133875499200524063689912560717606
0588611646710940507754100225698315520005593572972571636269561882670428252483600823257530420752963450

Find the thirteen adjacent digits in the 1000-digit number that have the greatest product. What is the value of this product?

Answer

The product is 23514624000 and the digits that got it are '5576689664895'

Idea

Pretty simple just substring all possible 13 long sub sections of the number. Multiply each digit in the string together return as PRODUCT. If the new product is greater then any product seen before overwrite the old one.

Python code

```
def prodOfDigits(numberString):
   product = 1
   for character in numberString:
      product = product * int(character)
   return product
def lrgstAdjacent(numberString, digits):
   largestProd = float('-inf')
   largestString = ""
   startIndex = 0
   while startIndex + digits <= len(numberString) - 1:
```

```
      newProd = prodOfDigits(numberString[startIndex : startIndex + digits])
      if newProd > largestProd:
         largestProd = newProd
         largestString = numberString[startIndex : startIndex + digits]

      startIndex = startIndex + 1
   return largestProd, largestString

input = "731671765313306249192251196744265747423553491949349698352031277450632623957831801698480186947885184385
print(lrgstAdjacent(input, 13))
```