```
In [10]:
           1 # coding for the closest value of the zero if there are so many values we ha
           2 n=int(input())
           3 s=input()
           4 s=s.split()
           5
             # "1 2 3 4 5"
             li=[]
           7
              for i in s:
                  li.append(abs(int(i)))
           8
           9
             min(li)
          10
          11
         5
         -1 2 4 7 3
Out[10]: 1
```

```
In [17]:
              # You have been given an integer array A of size You have to print the value
              # to print the greater value
           2
           3
           4
              n=int(input())
           5
              s=input()
           6
              s=s.split()
              # "1 2 3 4 5"
           7
           8
              li=[]
              for i in s:
           9
                   li.append(abs(int(i)))
          10
          11
              #for i in range (0,max(li)):
          12
          13
              def closestZero(li):
          14
          15
                   if 0 in li:
          16
                       return 0
          17
                   else:
          18
                       li.sort()
          19
                       pc = []
          20
                       nc= []
                       for i in li:
          21
          22
                           if i>0:
                               pc.append(i)
          23
                       for i in li:
          24
                           if i<0:
          25
          26
                               nc.append(i)
          27
                       if len(nc)==0:
                           return min(pc)
          28
          29
                       elif len(pc)==0:
          30
                           return max(nc)
          31
                       else:
          32
                           nz=max(nc)
          33
                           pz=min(pc)
          34
                           if abs(nz) > pc or abs(nc)==pc:
          35
                               return pc
          36
                           else:
          37
                               return nz
          38
              closestZero(li)
          39
          40
          41
          -1 1 2 -2 300 -300
Out[17]: 1
In [ ]:
           1
```

```
localhost:8888/notebooks/Desktop/problemsolvingprogramming-june-2019/13-june-2019.ipynb
```

```
In [1]:
             # Function to find the no of hours in a given time period
          1
          2
          3
             def isLeapYear(year): # To check if a given year is a Leap Year
                 if year % 400 == 0 or (year % 100 != 0 and year % 4 == 0):
          4
          5
                      return True
          6
                 return False
          7
          8
          9
         10
         11
             def numberOfDays(startyear, endyear):
                 sum = 0
         12
         13
                 for year in range(startyear, endyear+1):
                      if isLeapYear(year):
         14
         15
                          sum = sum + 366
         16
                      else:
         17
                          sum = sum + 365
         18
                 return sum
             #number of days in middle years of 2016 2019
         19
             numberOfDays(2017, 2018)
         20
         21
         22
             def numberOfDaysMonth(month, year):
                 if month == 2:
         23
         24
                      if isLeapYear(year):
         25
                          return 29
         26
                      return 28
                 elif (month <= 7 and month % 2!= 0) or (month >= 8 and month % 2 == 0):
         27
         28
                      return 31
         29
                 else:
         30
                      return 30
         31
             def daysInStartYear(startmonth, startyear):
         32
                 days = 0
         33
         34
                 for month in range(startmonth, 13):
                      days += numberOfDaysMonth(month, startyear)
         35
         36
                 return days
         37
             def daysInEndYear(endmonth, endyear):
         38
         39
                 days = 0
         40
                 for month in range(1, endmonth+1):
         41
                      days += numberOfDaysMonth(month, endyear)
         42
                 return days
         43
             def numberOfHours(startmonth, startyear, endmonth, endyear):
         44
         45
                 days = 0
         46
                 if startyear != endyear:
         47
                      days += daysInStartYear(startmonth, startyear)
                      days += daysInEndYear(endmonth, endyear)
         48
                      if endyear - startyear == 2: # 2019 - 2017
         49
                          days += numberOfDays(startyear+1, startyear+1)
         50
         51
                      elif endyear - startyear > 2:
         52
                          days += numberOfDays(startyear+1, endyear-1)
         53
                 else:
         54
                      for month in range(startmonth, endmonth+1):
                          days += numberOfDaysMonth(month, startyear)
         55
                 return 24 * days
         56
```

57 58 numberOfHours(11, 1975,3, 2018)

Out[1]: 371808

In []: 1 P