Nearest Smaller to Left.

Given an array of integers find the closest (w.r.t Considering distance, but value) smaller on left of every element. If an element has no smaller on the left side, print -1.

arr[]:

4	5	2	10	8
-1	4	-1	2	2

Brute force:-

```
for (int i=0; i<n; i++)
    for (int j=i-1; j>=0; j--)
```

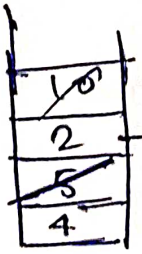
int j=(i-1)

↳ related to i i.e stack will be applied.

arr[]:

4	5	2	10	8

i i



S.top()

Stack

vector

-1	4	-1	2	2

- 4 :- stack is empty Push -1.
- is 4 smaller than 5 & yes put in vector.
- S.top < arr[i] Push stop
- S.top > arr[i] pop element.

• Here also we need not to reverse.

Code.

```
vector <int> v;
Stack <int> s;

for (int i=0; i<size; i++)
{
   if (s.size() == 0)
     {
       v.push_back(-1);
     }
   else if (s.size() > 0 && s.top() < arr[i])
     {
       v.push_back(s.top())
     }
   else if (s.size() > 0 && s.top >= arr[0])
   {
     while(s.size > 0 && s.top() >= arr[i])
       { s.pop();
       }
     if(s.size() == 0)
       v.push_back(-1)
     else
       v.push_back(s.top());
   }
   s.push_back(arr[i])
}
```