

```

141
142 // get longest palindrome subArray
143 pair<int , int> getLongestPalindrome(){
144     int oddIndex = 0 , evenIndex = 0;
145     for (int i = 0; i < n; i++){
146         if(d1[i] > d1[oddIndex]) oddIndex = i;
147         if(d2[i] > d2[evenIndex]) evenIndex = i;
148     }
149     int l1 = evenIndex-d2[evenIndex] , r1 = l1+(d2[evenIndex]*2)-1;
150     int l2 = oddIndex-d1[oddIndex]+1 , r2 = l2+(d1[oddIndex]*2-1)-1;
151     return r1-l1+1 ≥ r2-l2+1 ? make_pair(l1 , r1) : make_pair(l2 , r2);
152 }
153 };

```