// MSS DAC Approach

```
#include <stdio.h>
#include <stdlib.h>
#include <limits.h>

int main()
{
    int A[] = {-2,-5,6,-2,-3,1,5,-6}; // Array Value
    int n = 8; // Array Index = 8
    int low = 0;
    int high = n-1;

    int maxCrossingSum(int A[], int l, int mid, int r){
        int sum = 0;
        int lsum = INT_MIN;
        for(int i=mid;i>=0;i--){
            sum=sum+A[i];
            if(sum>lsum){
                lsum = sum;
```

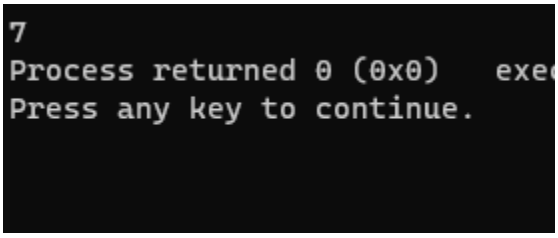
```
        }
    }
    sum = 0;
    int rsum = INT_MIN;
    for(int i = mid+1; i<=r;i++){
        sum=sum+A[i];
        if(sum>rsum){
            rsum = sum;
        }
    }
    return (lsum+rsum);
}

int maxSubarraySum(int A[], int low, int high){
    if(low == high){
        return A[high];
    } // Checking if index have more then one value
    else{
        int mid = low + (high - low)/2;
        int left_sum = maxSubarraySum(A,low,mid);
```

```
        int right_sum = maxSubarraySum(A,mid+1,high);
        int crossing_Sum =
maxCrossingSum(A,low,mid,high);
        return (left_sum,right_sum,crossing_Sum);
    }
}

int result = maxSubarraySum(A,low,high);
printf("%d",result);

return 0;
}
// Output:
```



// Time Complexity: (nlog n)