

Project Report

Bannuru Rohit Kumar Reddy : 21CS30011

Dumpala Hashmitha : 21CS10023

\

Problem Statement :

Given an array of n elements , we have to find the maximum continuous subarray sum.

For that , we will calculate maximum subarray sum and minimum subarray sum in a normal array

final result =max(maximum subarray sum, total array sum-minimum subarray sum)

Algorithm:

Function to calculate both maximum subarray and minimum subarray

:

Initialize variable 'total_sum' to 0 , 'cur_maxsum' to 0 and variable 'max_sum' to INT_MIN, and 'cur_minsum' to 0 and 'min_sum' to INT_MAX

Run a loop in the array 'a' from i=0 to i=size-1

total_sum=total_sum+a[i]

update cur_maxsum as, cur_maxsum=cur_maxsum+a[i]

update cur_minsum as, cur_minsum=cur_minsum+a[i]

update max_sum as, max_sum=max(max_sum,cur_maxsum)

update min_sum as, min_sum=min(min_sum,cur_minsum)

if(cur_maxsum<0)

update cur_maxsum=0

if(cur_minsum>0)

update cur_minsum=0

After the loop ends , we will get the final values of max_sum and min_sum

Now, final result =maximum of (max_sum) and (total_sum-min_sum)

MIPS implementation:

We will store cur_maxsum in \$s2, max_sum in \$s3, cur_minsum in \$s4 , min_sum in \$s5, total_sum in \$s6
then we will do ,li \$s3, -9999999 and li \$s5, 9999999 , assuming those are the max values of array elements the user can input

Then will take input n , which is the number of elements of the array
Then we will run a loop to take the input of n numbers
we will store n in \$s1 and then in the loop update values as discussed in the above algorithm

Example :

N = 7

Array = 8 -8 9 -9 10 -11 12

Total Linear sum : 11

Maximum Linear sum : 12

Minimum linear sum : -11

final sum of circular sum = Max (12, (11- -(11))) = 22