Για την λύση της άσκησης θα χρησιμοποιήσουμε τον αλγόριθμο του Kadane. Σε ένα array n στοιχείων βρίσκουμε το subarray με το μεγαλύτερο άθροισμα χρησιμοποιώντας 2 μεταβλητές.

Δηλώνουμε δύο μεταβλητές local sum (που δηλώνει το max sum στην θέση (n) του πίνακα) και global sum (που δηλώνει το max sum που έχουμε βρει έως τώρα).

Με μία for υπολογίζουμε σε κάθε θέση το local sum συγκρίνοντας το value της θέσης (n) με το άθροισμα του value της θέσης (n) και το value της θέσης (n -1) και η μεγαλύτερη τιμή είναι η νέα local sum. H global sum παίρνει μία αρχική minimum value και μετά ενημερώνεται κάθε φορά που η local sum > global sum όπου παίρνει την τιμή της local sum.

Για παράδειγμα, στον πίνακα {-1, 2, 3, -4, 8, 2, -3, 1, -4}

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INDEX POSITION | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| LM = LOCAL MAXIMUM |  |  |  |  |  |  |  |  |  |
| GM = GLOBAL MAXIMUM |  |  |  |  |  |  |  |  |  |
| LM = GM COLOR CODE |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| POSITION VALUE | -1 | 2 | 3 | -4 | 8 | 2 | -3 | 1 | -4 |
| ΠΡΩΤΗ ΘΕΣΗ ΤΟΥ subarray ΟΤΑΝ Η LM = GM = VALUE ΤΟΥ INDEX POSITION. ΣΤΗ ΘΕΣΗ 0 ΤΟΥ array ΙΣΧΥΕΙ ΟΠΟΤΕ Η ΠΡΩΤΗ ΘΕΣΗ ΞΕΚΙΝΑΕΙ ΑΠΟ ΤΗ ΘΕΣΗ 0 ΚΑΙ ΣΤΗ ΣΥΝΕΧΕΙΑ ΑΝΤΙΚΑΘΙΣΤΑΤΑΙ ΑΠΟ ΤΗ ΘΕΣΗ 1 ΤΟΥ array ΚΑΘΩΣ Η ΣΥΝΘΗΚΗ ΙΣΧΥΕΙ ΚΑΙ ΠΑΛΙ ΜΕΣΑ ΣΤΗΝ FOR |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  | -4 |
|  |  |  |  |  |  |  | 1 | -3 |
|  |  |  |  |  |  | -3 | -2 | -6 |
|  |  |  |  |  | 2 | -1 | 0 | -4 |
|  |  |  |  | 8 | 10 | 7 | 8 | 4 |
|  |  |  | -4 | 4 | 6 | 3 | 4 | 0 |
|  |  | 3 | -1 | 7 | 9 | 6 | 7 | 2 |
|  |  | 2 | 5 | 1 | 9 | 11 | 8 | 9 | 5 |
| ΤΕΛΕΥΤΑΙΑ ΘΕΣΗ ΤΟΥ subarray ΟΤΑΝ LM = GM. ΤΕΛΕΥΤΑΙΑ ΦΟΡΑ ΠΟΥ ΣΥΝΑΝΤΑΤΑΙ ΑΥΤΗ Η ΣΥΝΘΗΚΗ ΜΕΣΑ ΣΤΗ FOR ΕΙΝΑΙ ΣΤΟ POSITION 5 ΟΠΟΤΕ ΑΥΤΟ ΕΙΝΑΙ ΚΑΙ ΤΟ ΤΕΛΕΥΤΑΙΟ INDEX POSITION ΓΙΑ ΤΗ ΔΗΜΙΟΥΡΓΙΑ ΤΟΥ subarray | -1 | 1 | 4 | 0 | 8 | 10 | 7 | 8 | 4 |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| LM = -1 | LM = 2 | LM = 5 | LM = 1 | LM = 9 | LM = 11 | LM = 8 | LM = 9 | LM = 5 |
| GM = -1 | GM = 2 | GM = 5 | GM = 5 | GM = 9 | GM = 11 | GM = 11 | GM = 11 | GM = 11 |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

Με αυτόν τον τρόπο πετυχαίνουμε μόνο με μία επανάληψη (for) να έχουμε το επιθυμητό αποτέλεσμα καθώς από κάθε position της for παίρνουμε το local maximum και το global maximum. Έτσι δεν χρειάζεται 2η ή 3η επανάληψη μειώνοντας τον χρόνο σε O(n) αντί σε Ο(n2) ή Ο(n3) καθώς ο χρόνος υπολογισμού εξαρτάται μόνο από το πόσα n στοιχεία υπάρχουν στον πίνακα εφόσον οι συγκρίσεις σε κάθε θέση γίνονται μεταξύ των δύο variables local max global max.