

# ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ

## ΑΛΓΟΡΙΘΜΟΙ ΚΑΙ ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

Άσκηση: Max Sub-array

Πάυλος Λουκαρέας

A.M. 1046970

### Εισαγωγή

Στην εργασία αυτή γίνεται η υλοποίηση των αλγορίθμων σε γλώσσα C της εύρεσης μέγιστου αθροίσματος υπο-πίνακα. Κάθε αρχείο πηγαίου κώδικα δομείται σε δύο συναρτήσεις. Την `main()` η οποία εκτελείται πρώτη, φτιάχνει τους δύο τυχαίους πίνακες και η οποία στη συνέχεια καλεί δύο φορές τη συνάρτηση `algo_()` η οποία δέχεται έναν πίνακα, τυπώνει το μέγιστο άθροισμα υπο-πίνακα, καθώς και το χρόνο εκτέλεσης που απαιτήθηκε κάθε φορά. Λόγω της στατικής κατανομής μνήμης της C στη δεύτερη και τρίτη έκδοση του προβλήματος καλούμε δύο φορές την ίδια συνάρτηση με στατική κατανομή των πινάκων. Παρακάτω για κάθε αρχείο κώδικα παρουσιάζουμε τον αλγόριθμο και τα αποτελέσματα του.

`algo_v1.c`

Στην υλοποίηση του αλγορίθμου τα δεδομένα εισάγονται σε τριπλό nested loop με τους μετρητές να φτάνουν έως το μέγεθος του πίνακα  $N$ . Οπότε θα έχουμε πολυπλοκότητα τάξης  $O(N^3)$  πράγμα που σημαίνει κακός χρόνος εκτέλεσης. Τα αποτελέσματα για  $N = 1000$  ήταν:

```
starting point: 457 ending point: 473
result: 64
time taken: 0.498000
starting point: 164 ending point: 253
result: 101
time taken: 3.840000
Press any key to continue . . . _
```

algo\_v2.c

Στην υλοποίηση του αλγορίθμου τα δεδομένα εισάγονται σε απλό και διπλό nested loop με τους μετρητές να φτάνουν έως το μέγεθος του πίνακα  $N$ . Οπότε θα έχουμε σαν συνολική πολυπλοκότητα, αυτή που κυριαρχεί, δηλαδή της τάξης  $O(N^2)$ , πράγμα που σημαίνει ικανοποιητικός χρόνος εκτέλεσης. Τα αποτελέσματα για  $N = 10000$  ήταν:

```
starting point: 6640 ending point: 7048
result: 168
time taken: 0.152000
starting point: 13141 ending point: 13688
result: 221
time taken: 0.589000

Press any key to continue . . . _
```

algo\_v3.c

Στην υλοποίηση του αλγορίθμου τα δεδομένα εισάγονται σε δύο απλά nested loop με τους μετρητές να φτάνουν έως το μέγεθος του πίνακα  $N$ . Οπότε θα έχουμε σαν συνολική πολυπλοκότητα της τάξης  $O(N)$ , πράγμα που σημαίνει πολύ καλός χρόνος εκτέλεσης. Ωστόσο σαν μειονέκτημα από αυτή την υλοποίηση είναι ότι δεν μπορούμε να βρούμε τους δείκτες αρχής και τέλους του υπο-πίνακα. Τα αποτελέσματα για  $N = 100000$  ήταν:

```
starting point: 0 ending point: 0
result: 298
time taken: 0.001000
starting point: 0 ending point: 0
result: 221
time taken: 0.002000

Press any key to continue . . .
```