**Εξασφάλιση Ποιότητας και Πρότυπα**

**3η Εργασία**

Ζητούμενο 1

Παρακάτω φαίνονται οι αναλυτικοί πίνακες για τις τρεις συνολικά ρουτίνες που διαθέτουν οι υλοποιήσεις Α και Β, στους οποίους παρουσιάζονται αναλυτικά οι διάφοροι τελεστές και τα έντελα, καθώς και το πλήθος των εμφανίσεών τους.

Παραδοχές:

1. Όσον αφορά στον πίνακα number[], τον αντιμετωπίζω ως μία “συνάρτηση” (αρά τελεστή), η οποία δέχεται ως είσοδο παραδείγματος χάριν μια μεταβλητή, k. Γι΄ αυτό ακόμα το λόγο δεν μετράω ως ξεχωριστό τελεστή τις αγκύλες “[]”.
2. H sort\_numbers\_ascending αποτελεί μια ρουτίνα, η οποία εξ’ ορισμού (μιας και αναφερόμαστε στη γλώσσα C) απαιτεί τα άγκιστρα “{}” καθώς και τις παρενθέσεις “()”. Γι΄αυτό το λόγο δεν τα μετράω ως ξεχωριστούς τελεστές.
3. Οι ελάχιστες υλοποιήσεις που αφορούν στη δομή ελέγχου if και στη δομή επανάληψης for και while δεν απαιτούν τα άγκιστρα “{}”, συνεπώς μετράω τα τελευταία ως ξεχωριστό τελεστή. Παρομοίως και για τις υπόλοιπες ρουτίνες της άσκησης.
4. Δεν προσμετρώ την εισαγωγή του header “#include <stdio.h>”.
5. Η εντολή printf (στη C) εμφανίζεται υποχρεωτικά με ένα ζεύγος παρενθέσεων “()”, συνεπώς δεν το μετράω ως ξεχωριστό τελεστή. Ωστόσο, μιας και η ελάχιστη υλοποίηση της παραπάνω εντολής είναι χωρίς κόμμα “,”, μετράω το τελευταίο ως ξεχωριστό τελεστή.
6. Αντίστοιχα, η εντολή for (στη C) εμφανίζεται υποχρεωτικά με ένα ζεύγος παρενθέσεων και δύο ερωτηματικά, τα παραπάνω επομένως δεν προσμετρώνται ξεχωριστά ως τελεστές.
7. Η εντολή scanf (στη C) εμφανίζεται υποχρεωτικά (ως η ελάχιστη υλοποίησή της) με ένα ζεύγος παρενθέσεων καθώς και ένα κόμμα “,” συνεπώς τα παραπάνω δεν προσμετρώνται ξεχωριστά ως τελεστές. Σημαντικό να αναφερθεί είναι πως το σύμβολο “&” δεν συνοδεύει πάντοτε την εντολή scanf, αφού αυτό εξαρτάται από τον τύπο του input που θα δώσει ο χρήστης, επομένως θεωρείται ξεχωριστός τελεστής.
8. Η εντολή while (στη C) εμφανίζεται υποχρεωτικά με ένα ζεύγος παρενθέσεων “()”, συνεπώς το τελευταίο δεν προσμετράται ως ξεχωριστός τελεστής.
9. Αυτό που ακολουθεί μετά το κόμμα “,” στην εντολή scanf θεωρώ πως αποτελεί ένα ενιαίο έντελο. Παραδείγματος χάριν, στην εντολή “**scanf("%d", &count);**”η έκφραση &count είναι ένα ενιαίο έντελο, συνεπώς στο ξεχωριστό έντελο count που υπάρχει παρακάτω στους πίνακες, δεν προσμετράται σε αυτή την έκφραση η μεταβλητή count ως εμφάνιση.

*Α’ υλοποίηση (sort\_numbers\_ascending)*

| | **Τελεστές** | **Αριθμός Εμφανίσεων** | | --- | --- | | void | 1 | | sort\_numbers\_ascending() {} | 1 | | int | 3 | | number[] | 8 | | for ( ; ; ) | 3 | | = | 6 | | < | 3 | | ++ | 3 | | {} | 3 | | if () | 1 | | ; | 6 | | + | 1 | | printf() | 2 | | , | 5 | | > | 1 | | **n1 = 15** | **N1 = 47** | | | **Έντελα** | **Αριθμός Εμφανίσεων** | | --- | --- | | count | 4 | | temp | 3 | | i | 5 | | j | 8 | | k | 7 | | 0 | 2 | | 1 | 1 | | "Numbers in ascending order:\n" | 1 | | "%d\n" | 1 | | **n2 = 9** | **N2 = 32** | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

*Α’ υλοποίηση (main)*

| | **Τελεστές** | **Αριθμός Εμφανίσεων** | | --- | --- | | void | 1 | | main() {} | 1 | | int | 1 | | , | 4 | | number[] | 2 | | = | 2 | | ; | 8 | | printf() | 3 | | scanf( , ) | 3 | | while () | 1 | | {} | 1 | | > | 1 | | for ( ; ; ) | 1 | | < | 1 | | ++ | 1 | | sort\_numbers\_ascending() | 1 | | **n1 = 16** | **N1 = 32** | | | **Έντελα** | **Αριθμός Εμφανίσεων** | | --- | --- | | i | 4 | | count | 3 | | 20 | 2 | | t | 2 | | 0 | 2 | | "How many numbers you are going to enter:" | 1 | | "%d" | 3 | | &count | 2 | | "\nEnter the numbers one by one:" | 1 | | "\nThis is a test" | 1 | | &number[i] | 1 | | **n2 = 11** | **N2 = 22** | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

*Β’ υλοποίηση (main)*

| | **Τελεστές** | **Αριθμός Εμφανίσεων** | | --- | --- | | void | 1 | | main() {} | 1 | | int | 2 | | num[] | 8 | | , | 8 | | = | 9 | | ; | 16 | | printf() | 6 | | scanf( , ) | 5 | | while () | 1 | | > | 2 | | {} | 5 | | for ( ; ; ) | 5 | | < | 5 | | -- | 1 | | ++ | 4 | | + | 1 | | **n1 = 17** | **N1 = 80** | | | **Έντελα** | **Αριθμός Εμφανίσεων** | | --- | --- | | i | 15 | | 20 | 4 | | t | 4 | | 0 | 4 | | n | 3 | | count | 3 | | j | 7 | | a | 3 | | x | 1 | | b | 1 | | "How many numbers you are going to enter:" | 1 | | "%d" | 5 | | &count | 4 | | "\nEnter the numbers one by one:" | 1 | | "\nThis is a test" | 1 | | "\nThis is my test" | 1 | | &num[i] | 1 | | "Numbers in ascending order:\n" | 1 | | "%d\n" | 1 | | **n2 = 19** | **N2 = 61** | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

Ζητούμενο 2

Σε αυτό το μέρος για κάθε μία από τις παραπάνω ρουτίνες υπολογίζονται ορισμένες μετρικές, σύμφωνα με την εκφώνηση. Προηγουμένως ωστόσο θα αναφερθούν τα μεγέθη που απαιτούνται για τον υπολογισμό των ζητουμένων[[1]](#footnote-0).

* Λόγος του εκτιμητή μήκους προς το μήκος προγράμματος (Νest/N): **Nest = n1 \* log2(n1) + n2 \* log2(n2)** και **Ν = Ν1 + Ν2**
* Επίπεδο Προγράμματος (L): **Lest = 2 \* n2 / n1 \* N2**
* Επίπεδο Γλώσσας (λ):
  + Λεξιλόγιο: n = n1 + n2
  + Όγκος: V = N \* log2(n)
  + Άρα, **λ = L2 \* V**
* Λόγος αριθμού γραμμών σχολίων προς τον αριθμό φυσικών γραμμών κώδικα: **C = γραμμές σχολίων / φυσικός αριθμός[[2]](#footnote-1)** κώδικα όπου φυσικός αριθμός κώδικα = # των γραμμών λαμβάνοντας υπόψιν σχόλια και κενές γραμμές

*Α’ υλοποίηση (sort\_numbers\_ascending)*

Υπενθυμίζεται ότι **n1 = 15, N1 = 47, n2 = 9** και **Ν2 = 32**

1. Nest/N
   1. Ν = 47 + 32 = 79
   2. Νest = 15 \* log2(15) + 9 \* log2(9) = 15 \* 3.906 + 9 \* 3.169 = 58.59 + 28.521 = 87.111
   3. Άρα, Nest/N = 87.111 / 79 = 1,102
2. L = Lest = 2 \* 9 / 15 \* 32 = 18 / 480 = 0.037
3. λ
   1. n = 15 + 9 = 24
   2. V = 79 \* log2(24) = 79 \* 4.584 = 362.136
   3. Άρα, λ = L2 \* V = 0.0372 \* 362.136 = 0.495
4. C[[3]](#footnote-2) = 1 / 17 = 0.058

*Α’ υλοποίηση (main)*

Υπενθυμίζεται ότι **n1 = 16, N1 = 32, n2 = 11** και **Ν2 = 22**

1. Nest/N
   1. Ν = 32 + 22 = 54
   2. Νest = 16 \* log2(16) + 11 \* log2(11) = 16 \* 4 + 11 \* 3.459 = 64 + 38.049 = 102.049
   3. Άρα, Nest/N = 102.049 / 54 = 1.889
2. L = 2 \* 11 / 16 \* 22 = 22 / 352 = 0.062
3. λ
   1. n = 16 + 11 = 27
   2. V = 54 \* log2(27) = 54 \* 4.754 = 256.716
   3. Άρα, λ = 0.0622 \* 256.716 = 0.986
4. C[[4]](#footnote-3) = 5 / 15 = 1 / 3 = 0.333

*Β’ υλοποίηση (main)*

Υπενθυμίζεται ότι **n1 = 17, N1 = 80, n2 = 19** και **Ν2 = 61**

1. Nest/N
   1. Ν = 80 + 61 = 141
   2. Νest = 17 \* log2(17) + 19 \* log2(19) = 17 \* 4.087 + 19 \* 4.247 = 69.479 + 80.693 = 150,172
   3. Άρα, Nest/N = 150,172 / 141 = 1.065
2. L = 2 \* 19 / 17 \* 61 = 38 / 1037 = 0.036
3. λ
   1. n = 17 + 19 = 36
   2. V = 141 \* log2(36) = 141 \* 5.169 = 728.829
   3. Άρα, λ = 0.0362 \* 728.829 = 0.944
4. C = 19 / 45 = 0.422

Ζητούμενο 3

1. Πραγματοποιείται στρογγυλοποίηση των δεκαδικών αριθμών έως και το τρίτο ψηφίο μετά την υποδιαστολή [↑](#footnote-ref-0)
2. Οι γραμμές κώδικα κάθε ρουτίνας μετρώνται μεταξύ των αγκίστρων “{}” που την περικλείουν [↑](#footnote-ref-1)
3. Όσον αφορά στην ρουτίνα sort\_numbers\_ascending λαμβάνεται υπόψιν το σχόλιο που βρίσκεται ακριβώς πάνω από το όνομά της αλλά όχι αυτό που βρίσκεται στην αρχή του προγράμματος, μιας και αυτό δεν αφορά την συγκεκριμένη ρουτίνα ξεκάθαρα [↑](#footnote-ref-2)
4. Θεωρώ πως το σχόλιο που βρίσκεται στην αρχή του προγράμματος αφορά στην συνάρτηση main [↑](#footnote-ref-3)