



3^η Εργασία

Διαδικαστικά

Η εργασία είναι **αυστηρά ατομική** και αποτελεί την 3^η από τις 5 εργασίες του μαθήματος. Ως 5^η εργασία θα υπολογιστεί η συμμετοχή στη διόρθωση μιας εργασίας. Τα διαδικαστικά που αφορούν τις εργασίες αναφέρονται αναλυτικά στις πληροφορίες του μαθήματος στο eClass. **Αντιγραφή σε κάποια εργασία συνεπάγεται μηδενισμό σε όλες τις εργασίες αυτού του έτους.**

Όλες οι εργασίες θα παραδοθούν αυστηρά μέσω eClass.

Η 3 εργασία έχει καταληκτική ημερομηνία και ώρα παράδοσης **Δευτέρα 15/12/2023** και ώρα **23:30** (πείτε στον εαυτό σας ότι το σύστημα κλείνει 10 το βράδυ και ότι η μισή ώρα είναι για να μην τύχει κάτι). Μετά βγείτε για ποτό! (σοβαρά τώρα έχετε 21 ημέρες για μια εύκολη σχετικά εργασία... Μην την αφήσετε για τη τελευταία Πέμπτη και μην την παραδώσετε στις 10 και κάτι!!!) **Καμία εργασία δεν θα γίνει δεκτή μετά τη λήξη της προθεσμίας¹.**

ΣΗΜΑΝΤΙΚΟ:

Για την εργασία παραδώστε μόνο ένα αρχείο pdf (π.χ. Xenos_Michalis.pdf) με το όνομά σας. Μέσα στο κείμενο δεν θα πρέπει να υπάρχει καμία πληροφορία για εσάς (ούτε όνομα, ούτε αριθμό μητρώου, ούτε τίποτε άλλο). Είναι κρίμα να χάνετε μονάδες έτσι άρα ελέγξτε τα αρχεία σας!

Ζητούμενο

Θέλουμε να γράψουμε κώδικα (ας υποθέσουμε σε python) για το μαθηματικό κόλπο που προτάθηκε από τον Ινδό μαθηματικό **D. R. Kaprekar**. Το κόλπο αφορά τετραψήφιους αριθμούς και πάντα οδηγεί στον ίδιο μυστηριώδη αριθμό, το 6174.

Για παράδειγμα:

1. Επιλέγετε έναν τετραψήφιο αριθμό, όπως το 2390. (Προσοχή: όλοι οι αριθμοί πρέπει να έχουν τουλάχιστον δύο διαφορετικά ψηφία, δηλαδή όχι 1111, 2222 κτλ.)
2. Ταξινομείτε τα ψηφία του αριθμού για να δημιουργήσετε:
 - τον μεγαλύτερο δυνατό αριθμό: 9320
 - τον μικρότερο δυνατό αριθμό: 0239 → 239
3. Υπολογίζετε τη διαφορά μεταξύ των δύο
4. Επαναλαμβάνετε τη διαδικασία με το νέο αποτέλεσμα
5. Μόλις φτάσετε στον αριθμό **6174**, η διαδικασία επαναλαμβάνεται για πάντα

Το αποτέλεσμα είναι (πάντα;) 6174!

¹ Αυτό είναι κάτι που το τηρώ αυστηρά και δεν θα παρεκκλίνω ποτέ, άρα μην στείλετε εργασία 23:35 με e-mail.



Τμήμα Μηχανικών Η/Υ & Πληροφορικής

Πανεπιστήμιο Πατρών

NE5577 Ποιότητα Λογισμικού

Επειδή μας φάνηκε πολύ απλό να εξηγήσουμε και να κωδικοποιήσουμε τον συγκεκριμένο μαθηματικό κανόνα, μπορούμε να γράψουμε ένα πρόγραμμα σε Python που θα εκτελεί αυτόματα το κόλπο για οποιονδήποτε τετραψήφιο αριθμό.

Επειδή μας φάνηκε πολύ απλό να γράψουμε τον συγκεκριμένο κώδικα, ζητήσαμε από το ChatGPT (τη δωρεάν έκδοση):

Create a program in ANSI C to ask a 3-digit number and perform the Einstein trick that resulting 1089.

Και πήραμε αυτή την απάντηση:

```
1  def kaprekar_routine(number):
2      count = 0
3      while number != 6174:
4          # Convert number to 4-digit string
5          num_str = f"{number:04d}"
6
7          # Create largest and smallest numbers from digits
8          largest = int("".join(sorted(num_str, reverse=True)))
9          smallest = int("".join(sorted(num_str)))
10
11         # Subtract
12         number = largest - smallest
13         count += 1
14
15         print(f"Step {count}: {largest} - {smallest} = {number}")
16
17         # Check for a trivial loop (all digits equal)
18         if number == 0:
19             print("This number reaches 0 and won't reach 6174.")
20             return
21
22     print(f"Kaprekar's constant reached in {count} steps!")
23
24 # Ask user for input
25 while True:
26     user_input = input("Enter a 4-digit number (at least two different digits): ")
27
28     if user_input.isdigit() and len(user_input) == 4:
29         if len(set(user_input)) == 1:
30             print("All digits are the same. Try again.")
31         else:
32             kaprekar_routine(int(user_input))
33             break
34     else:
35         print("Invalid input. Enter exactly 4 digits.")
```

Στη συνέχεια ζητήσαμε από το ChatGPT:

Calculate Hallstead metrics in this code

Και απάντησε:



1 Python Version

Step 1: List of Operators and Operands

Operators (distinct, $n1$):

`=, +, -, !=, :, (), >, if, while, print, int(), set(), len(), break, and` → 15

Operands (distinct, $n2$):

`number, num_str, largest, smallest, count, user_input, 6174, 0, 4` → 9

- Vocabulary $n = n1 + n2 = 15 + 9 = 24$

Occurrences (total):

- Operators ($N1$) = 38
- Operands ($N2$) = 25
- Program length $N = N1 + N2 = 63$

Step 2: Halstead Metrics

Volume:

$$V = N \times \log_2(n) = 63 \times \log_2(24) \approx 63 \times 4.585 \approx 288.9$$

Difficulty:

$$D = \frac{n1}{2} \times \frac{N2}{n2} = \frac{15}{2} \times \frac{25}{9} = 7.5 \times 2.7778 \approx 20.83$$

Effort:

$$E = D \times V \approx 20.83 \times 288.9 \approx 6018$$

Language level:

$$\lambda = \frac{V}{E} \approx \frac{288.9}{6018} \approx 0.048$$

Ερώτημα 1: Μπορείτε να βρείτε πού έχει κάνει λάθος το ChatGPT και να δώσετε τους σωστούς υπολογισμούς και αποτελέσματα για τις τιμές αυτές; Καλύτερα δώστε ένα αναλυτικό πίνακα (όπως στο παράδειγμα στις διαφάνειες) που να περιλαμβάνει τα πάντα.

Στη συνέχεια και μετά από απαίτηση του κοινού ζητήσαμε από το ChatGPT να μας δώσει και μία εναλλακτική λύση και για τους συναδέλφους που δήλωσαν ότι θα προτιμούσαν η εργασία να είναι σε Anci C. Έτσι ζητήσαμε νέα υλοποίηση:

Great, but do the same for ANSI C, as my colleagues requested.

Και πήραμε ως απάντηση τον κώδικα:



Τμήμα Μηχανικών Η/Υ & Πληροφορικής
Πανεπιστήμιο Πατρών
NE5577 Ποιότητα Λογισμικού

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  void kaprekar_routine(int number) {
5      int count = 0;
6
7      if (number % 1111 == 0) {
8          printf("All digits are the same. This number will reach 0 and not 6174.\n");
9          return;
10     }
11
12     while (number != 6174) {
13         int digits[4];
14         int i;
15
16         // Extract digits
17         digits[0] = number / 1000;
18         digits[1] = (number / 100) % 10;
19         digits[2] = (number / 10) % 10;
20         digits[3] = number % 10;
21
22         // Sort digits ascending (simple bubble sort)
23         for (i = 0; i < 3; i++) {
24             int j;
25             for (j = i + 1; j < 4; j++) {
26                 if (digits[i] > digits[j]) {
27                     int temp = digits[i];
28                     digits[i] = digits[j];
29                     digits[j] = temp;
30                 }
31             }
32         }
33
34         // Form smallest number
35         int smallest = digits[0]*1000 + digits[1]*100 + digits[2]*10 + digits[3];
36         // Form largest number
37         int largest = digits[3]*1000 + digits[2]*100 + digits[1]*10 + digits[0];
38
39         number = largest - smallest;
40         count++;
41
42         printf("Step %d: %d - %d = %d\n", count, largest, smallest, number);
43     }
44
45     printf("Kaprekar's constant 6174 reached in %d steps!\n", count);
46 }
47
48 int main() {
49     int number;
50
51     printf("Enter a 4-digit number (at least two different digits): ");
52     scanf("%d", &number);
53
54     if (number < 1000 || number > 9999) {
55         printf("Invalid input. Enter exactly 4 digits.\n");
56     } else {
57         kaprekar_routine(number);
58     }
59
60     return 0;
61 }
```



Και ζητώντας να υπολογίσει ξανά τις μετρικές του Halstead μας έδωσε την παρακάτω απάντηση:

2 C Version

Step 1: List of Operators and Operands

Operators (distinct, n_1):

`=, +, -, *, /, %, !=, <, >, ==, (), {}, [], :, ,, if, else, while, for, printf, scanf` → 19

Operands (distinct, n_2):

`number, count, digits, i, j, temp, largest, smallest, 6174, 0, 1, 3, 4, 1000, 100, 10` → 16

- Vocabulary $n = n_1 + n_2 = 19 + 16 = 35$

Occurrences (total):

- Operators (N_1) = 55
- Operands (N_2) = 42
- Program length $N = N_1 + N_2 = 97$

Step 2: Halstead Metrics

Volume:

$$V = N \times \log_2(n) = 97 \times \log_2(35) \approx 97 \times 5.13 \approx 497.6$$

Difficulty:

$$D = \frac{n_1}{2} \times \frac{N_2}{n_2} = \frac{19}{2} \times \frac{42}{16} = 9.5 \times 2.625 \approx 24.94$$

Effort:

$$E = D \times V \approx 24.94 \times 497.6 \approx 12,398$$

Language level:

$$\lambda = \frac{V}{E} \approx \frac{497.6}{12,398} \approx 0.040$$

Ερώτημα 2: Μπορείτε να βρείτε πού έχει κάνει λάθος το ChatGPT και να δώσετε τους σωστούς υπολογισμούς για τις τιμές αυτές στη 2^η έκδοση του προγράμματος; Καλύτερα δώστε δύο αναλυτικούς πίνακες (όπως στο παράδειγμα στις διαφάνειες) ένα για τη συνάρτηση `isPalidrome()` και ένα για το `main()` και μετά υπολογίστε τα αποτελέσματα συνολικά. Η λύση σας πρέπει να περιλαμβάνει τα πάντα εκτός από τις γραμμές `#include`

Τέλος ζητήσαμε από το ChatGPT να συγκρίνει τις 2 εκδόσεις. Αυτό που ρωτήσαμε ήταν:

[can you compare the 2 versions of the codes based on the Halstead metrics results?](#)

Και η απάντηση που πήραμε ήταν:



Comparison:

- **Program Length (N):** The C code has a higher program length (97) compared to the Python code (63), indicating it contains more operators and operands and is overall longer.
- **Program Vocabulary (n):** The C code has a larger program vocabulary (35) than Python (24), showing it uses more unique operator and operand types.
- **Program Volume (V):** The C code has a higher volume (497.6) than the Python code (288.9), indicating it is a larger and more complex program in terms of information content.
- **Program Difficulty (D):** The C code has a higher difficulty (24.94) than the Python code (20.83), suggesting it may be harder to understand due to more low-level operations and explicit handling of details.
- **Language Level (λ):** The language level is slightly higher for Python (0.048) compared to C (0.040), indicating Python is slightly easier to work with and requires less effort for development.

These metrics suggest that while the C code has higher complexity, volume, and difficulty, it also contains more unique elements and requires more effort to write and maintain. The Python version is shorter, simpler, and easier to understand, highlighting the advantages of higher-level language features such as built-in functions and abstracted operations.

Ερώτημα 3: Μπορείτε να βρείτε πού έχει κάνει λάθος το ChatGPT σε αυτές τις συγκρίσεις; Δώστε τη δική σας αιτιολόγηση.

Checklist πριν την παράδοση

- Υποβάλλω μόνο ένα αρχείο pdf;
- Το pdf αρχείο μου είναι μικρότερο από 2Mb (θα ήταν αστείο να είναι μεγάλο αφού δεν ζητάμε εικόνες, αλλά αν τυχόν έχετε βάλει εικόνες -που δεν χρειάζεται- μην έχετε μέγεθος μεγαλύτερο από 2Mb);
- Έχω παραδώσει τη λύση και όχι την εκφώνηση; (συνηθισμένο λάθος)
- Έχω παραδώσει τη σωστή εργασία; (συνήθως γίνεται από τη 3^η εργασία και μετά κάποιος να μας στέλνει την προηγούμενη!);
- Μπορώ από το eClass να ανοίξω το pdf και να το κατεβάσω στο δίσκο μου; Είναι το σωστό pdf;
- Μέσα στο pdf έχω απαντήσει όλα τα ζητούμενα (π.χ. έχω βάλει τους πίνακες που ζητάει η εκφώνηση και έχω απαντήσει και τα 3 ερωτήματα);
- Έχω κείμενο που να εξηγώ μόνο ότι χρειάζεται;
- Έχω μήπως ξεχάσει κάπου το όνομά μου στο κείμενο; Αν ναι να το σβήσω!

Η ομάδα διόρθωσης θα έχει οδηγίες, αλλά σε αυτή τη φάση θα αποκαλύπταν σημεία της λύσης και δεν δίνονται.

Η ομάδα διόρθωσης θα κληρωθεί από όσους παρέδωσαν την εργασία.

Disclaimer: Ο εκτιμώμενος χρόνος που θα χρειαστεί κάποιος που παρακολούθησε παράδοση και φροντιστήριο για να τη λύσει άριστα δεν πρέπει να ξεπερνά τις 6-12 ώρες το πολύ! ΜΗΝ την αφήσετε όμως για τελευταία στιγμή και **ΜΗΝ εμπλακείτε σε διαδικασίες που μπορεί να σας φέρουν σε δύσκολη θέση.**