

# Εισαγωγή στον Προγραμματισμό

(CEID\_NY131)



## Οργάνωση Εβδομαδιαίων Δραστηριοτήτων Μαθήματος

Kleanthis Thramboulidis  
Prof. of Software and System Engineering  
University of Patras  
<https://sites.google.com/site/thramboulidiskleanthis/>



## Week 1 - Αντικείμενο

- Σημασία, Στόχος και Οργάνωση του Μαθήματος
  - Εργαστήριο (Το βασικότερο συνθετικό του Μαθήματος)
- Βασικές έννοιες Διαδικαστικού Προστακτικού Προγραμματισμού
  - με τη χρήση παραδειγμάτων από την καθημερινή ζωή
- Αφαιρετικότητα στις Διεργασίες (procedural abstraction)
  - Όνομα – Περιγραφή Διεργασίας
- Διαδικασία Ανάπτυξης Προγράμματος
  - Βασικά Βήματα (Λεκτική Περιγραφή, Η μετατροπή σε Πρόγραμμα)
  - Ο ρόλος του IDE στην διαδικασία αυτή
  - Οργάνωση αποθηκευτικού χώρου
  - Το πρόγραμμα HelloWorld

## Week 1 - Το βασικό Εργαλείο (EA-1)

Εργαστήριο 1ης εβδομάδας (Αντικείμενο) [UPDATED]

Δευτέρα, 30 Σεπτεμβρίου 2024 - 6:14 μ.μ. - από τον χρήστη Θραμπουλίδης Κλεάνθης

**Αντικείμενο 1<sup>ου</sup> Εργαστηρίου:** Εισαγωγή στο περιβάλλον του εργαστηρίου

Το 1ο εργαστήριο έχει στόχο τη γνωριμία του χώρου διεξαγωγής καθώς και του υπολογιστικού περιβάλλοντος που θα χρησιμοποιούμε κατά τη διάρκεια του εξαμήνου.

**Εισαγωγή στο περιβάλλον του εργαστηρίου**

### Δράσεις της EA-1

- Αξιοποίηση του Νέφους ως αποθηκευτικού χώρου
- Οργάνωση αποθηκευτικού χώρου
- Εγκατάσταση DevC++
- Διαδικασία Ανάπτυξης Προγράμματος με τη χρήση IDE
- Η περίπτωση του DevC++
- Το πρόγραμμα HelloWorld

## Week 1 - Παραδοτέο EA-1

Τίτλος

Αξιολόγηση 1ης Εβδομάδας

Ανάθεση σε: Συγκεκριμένες Ομάδες Χρηστών

Βαθμολογήστε από 0 (καθόλου) έως 5 (πάρα πολύ) τα παρακάτω ως προς το πόσο σας βοήθησαν για την επιτυχή εκτέλεση του 1<sup>ου</sup> Εργαστηρίου.

Δράσεις 1ης Εβδομάδας

Ανάθεση σε: Συγκεκριμένες Ομάδες Χρηστών

Το ερωτηματολόγιο είναι το παραδοτέο σας για το εργαστήριο της 1<sup>ης</sup> εβδομάδας και ως εκ τούτου η συμπλήρωσή του είναι υποχρεωτική.

**Αποτελέσματα  
Ερωτηματολογίου**

## Week 2 - Agenda 1/3 (Add2Numbers Version1)

Ένα πιο σύνθετο πρόγραμμα (**Add2Numbers Version1**)

- Μετατροπή Λεκτικής περιγραφής σε πηγαίο κώδικα
- Δημιουργία Project Add2Numbers
- Είσοδος δεδομένων (η συνάρτηση scanf)
- Η σημασία της Μεταβλητής
- Η σημασία των τύπων δεδομένων
- Οι Τελεστές ως βασικές διεργασίες
- Έκφραση
- Δομή ενός C προγράμματος

## Week 2 - Agenda 2/3 (Add2Numbers Version2)

Η δική μας πρώτη συνάρτηση (**getNumber()**)

- **Δήλωση** Συνάρτησης
- **Ορισμός** Συνάρτησης
- **Κλήση** Συνάρτησης

Μια πιο σύνθετη Συνάρτηση (**sum()**)

- Η έννοια της επαναχρησιμοποίησης (reusability)
- Ορίσματα συνάρτησης
- Επιστρεφόμενη τιμή

## Week 2 - Agenda 3/3 (HS - Fibonacci numbers)

### Ανάθεση έργου στη Μηχανή – Παραδείγματα

- **Δύο πιο σύνθετα προβλήματα**
  - Fibonacci numbers
  - Hailstone sequence
- **Hailstone sequence** (Λεκτική περιγραφή – 3 εκδοχές)
- Η επικοινωνία με την μηχανή με την χρήση της ακολουθίας αριθμών Fibonacci (Fibonacci numbers)
- Συγγραφή πηγαίου κώδικα

## Week 3 - Week 2 Αντικείμενο

- Μετατροπή Λεκτικής περιγραφής σε πηγαίο κώδικα (Add2Numbers)
- Δημιουργία Project Add2Numbers – Οργάνωση αποθηκευτικού χώρου
- Είσοδος δεδομένων (η συνάρτηση scanf)
- Η σημασία της Μεταβλητής
- Η σημασία των τύπων δεδομένων
- Οι Τελεστές ως βασικές διεργασίες
- Έκφραση
- Συνάρτηση ως αυτόνομη επαναχρησιμοποιήσιμη μονάδα κώδικα
- Δομή ενός C προγράμματος

Όλα τα παραπάνω τα είδαμε μέσα από ένα πιο σύνθετο πρόγραμμα (**Add2Numbers Version1, Version2, Version3**)

## Week 3 - Week 2 Το βασικό Εργαλείο (EA-2)

Εργαστήριο 1ης εβδομάδας (Αντικείμενο) [UPDATED]

Δευτέρα, 30 Σεπτεμβρίου 2024 - 6:14 μ.μ. - από τον χρήστη Θραμπουλίδης Κλεάνθης

**Αντικείμενο 1<sup>ο</sup> Εργαστηρίου:** Εισαγωγή στο περιβάλλον του εργαστηρίου

Το 1ο εργαστήριο έχει στόχο τη γνωριμία του χώρου διεξαγωγής καθώς και του υπολογιστικού περιβάλλοντος που θα χρησιμοποιούμε κατά τη διάρκεια του εξαμήνου.

**add2Numbers**

### Δράσεις της EA-2

- Αξιοποίηση του Νέφους ως αποθηκευτικού χώρου
  - Οργάνωση αποθηκευτικού χώρου
- Δημιουργία project για κάθε πρόγραμμα
- Διαδικασία Ανάπτυξης Προγράμματος με τη χρήση IDE
  - Η περίπτωση του DevCpp
- Ορισμός δικών μας συναρτήσεων

Κλεάνθης Θραμπουλίδης

Week Agenda

9

## Week 3 - Week 2 Παραδοτέο EA-2

### 1<sup>ο</sup> Μέρος

Όνομα Άσκησης	Ρυθμίσεις άσκησης	Αποτελέσματα
Παραδοτέο Εργαστηρίου 2ης Εβδομάδας (Μέρος 1ο) (έχει λήξει) Όλες οι ερωτήσεις της άσκησης αφορούν υλικό που έχουμε δουλέψει μέχρι σήμερα. Πριν από την εκτέλεση της άσκησης διαβάστε προσεκτικά τις Οδηγίες εκτέλεσης άσκησης στο eclass	Έναρξη: 11/10/24, 5:00 μ.μ. Λήξη: 13/10/24, 11:59 μ.μ. Διάρκεια: 20 λεπτά Προσπάθειες: 1 Προσωρινή αποθήκευση: Ναι	Εμφάνιση 85 υποβολές

### 2<sup>ο</sup> Μέρος

Τίτλος	Ημερομηνία	Απαντήσεις
Παραδοτέο Εργαστηρίου 2ης Εβδομάδας (Μέρος 2ο) Ανάθεση σε: Συγκεκριμένες Ομάδες Χρηστών	Από: Παρασκευή, 11 Οκτωβρίου 2024 - 7:00 μ.μ. έως: σήμερα - 2:00 π.μ.	59

**Αποτελέσματα  
Ερωτηματολογίου**

**Εργασία Ομάδας 2: (Σε 40 παραδοτέα) V3-20, V2-9, V1-4, Fail-7**

Κλεάνθης Θραμπουλίδης

Week Agenda

10

## Week 3 - Week 3 Αντικείμενο

- Η επικοινωνία με τη μηχανή – Η σημασία της Αφαιρετικότητας
- Προτάσεις Επανάληψης
- Μια πιο αφαιρετική αναπαράσταση Διεργασίας και η σημασία της
- Διαδικασία Ανάπτυξης Προγράμματος (Updated)
- Fibonacci numbers (Ανάπτυξη προγράμματος)
- Εισαγωγή στους Πίνακες (Arrays)
- Η διεργασία ως επαναχρησιμοποιήσιμη μονάδα κώδικα (συνάρτηση)
- Υπολογισμός μέσου όρου (Average) (Ανάπτυξη προγράμματος)
- Πίνακες ως όρισμα σε συνάρτηση
  - Η Συνάρτηση average()

## Τι κάνουμε αν έχουμε ερωτήσεις;

- Δείτε πρώτα από όλα αν η ερώτηση σας έχει ήδη απαντηθεί (**FAQs.pdf**)
- Εκφράστε την ερώτηση σας δημόσια.
  - Πιθανότατα αφορά και τους συμφοιτητές σας.
- ~~Για άμεση επικοινωνία χρησιμοποιήστε την ομάδα στο FB.~~
- Εναλλακτικά χρησιμοποιείτε την περιοχή συζητήσεων του eclass
- Αξιοποιήστε την δια ζώσης επικοινωνία στις διαλέξεις, τα διαλείμματα, το φροντιστήριο, το εργαστήριο.
- Αν η ερώτηση σας είναι προσωπική επικοινωνήστε με τους υπευθύνους του μαθήματος.

Παρακαλώ πολύ **ερωτήσεις με email μόνο όταν έχετε εξαντλήσει τους παραπάνω τρόπους**. Στην περίπτωση αυτή βάλτε στον τίτλο του μηνύματος την έκφραση **12P-Urgent** Μόνο σε αυτές τις περιπτώσεις θα περιμένετε απάντηση από εμένα. Οι προσωπικές σας απόψεις για το μάθημα ανήκουν στην κατηγορία αυτή και είναι ευπρόσδεκτες.

## Week 4 - Week 3 Παραδοτέο ΕΑ-3

Τίτλος	1 <sup>ο</sup> Μέρος	Υποβλ.			
Παραδοτέο Εργαστηρίου 3ης εβδομάδας (Μέρος 1ο)	<b>Turnitin</b>	79			
Ατομική εργασία	Title	Uploaded	Viewed	Grade	Similarity
Ανάθεση σε: Συγκεκριμένες Ομάδες Χρηστών	hailstone versions	October 21, 2024			81%
	HailstoneSequence (V1.2, ...	October 20, 2024			98%
	askisi ergastiriou	October 20, 2024			38%
	ΠΑΡΑΔΟΤΕΟ.pdf	October 20, 2024			68%
	Hailstone neofitos spyrid...	October 20, 2024			50%
Όνομα Άσκησης	2 <sup>ο</sup> Μέρος	Ρυθμίσεις άσκησης	Αποτελέσματα		
Παραδοτέο Εργαστηρίου 3ης Εβδομάδας (Μέρος 2ο) (έχει λήξει)	Όλες οι ερωτήσεις της Άσκησης αφορούν υλικό που έχουμε δουλέψει μέχρι σήμερα. Για τον λόγο αυτό εκτελέστε την άσκηση μετά από την προσεκτική μελέτη	Έναρξη: 19/10/24, 10:00 π.μ. Λήξη: 20/10/24, 11:59 μ.μ. Διάρκεια: 35 λεπτά	Εμφάνιση <b>81 υποβολές</b>		
		10<20min, 18<25min, 32<30min			
<i>Κλεάνθης Θραμπουζίδης</i>	Week Agenda	13			

## Χαρακτηριστικές Απαντήσεις 1/2

Δώστε την πρόταση για την εμφάνιση στην οθόνη του 3ου στοιχείου του πίνακα που δηλώνεται από την πρόταση `int fbn[20]`.

### Απάντηση

```
printf("Το 3ο στοιχειο του πινακα ειναι:%d\n", fbn[20]);
```

[Fibonacci Numbers] Ποια από τις δύο προτάσεις ελέγχου ροής προγράμματος `for` και `while` θα επιλέγατε και γιατί;

### Απάντηση

Θα χρησιμοποιούσα την πρόταση `while` επειδή είναι πιο ευελκτική εφόσον δεν χρειάζεται να γνωρίζουμε ακριβώς το χρονικό διάστημα για το ποιο θα διακοσσει η εκτέλεση του προγράμματος.

Δώστε τη δήλωση μιας συνάρτησης που δέχεται ένα πίνακα αριθμών κινητής υποδιαστολής απλής ακρίβειας και επιστρέφει το άθροισμα των περιττής τάξης στοιχείων του.

### Απάντηση

[Hailstone Sequence] Δράση [Δ3]: Τι προσθέσατε στον κώδικα της V1.1 για να πάρετε αυτόν της V1.2;

### Απάντηση

*Κλεάνθης Θραμπουζίδης*

Week Agenda

## Χαρακτηριστικές Απαντήσεις 2/2

Δώστε τη δήλωση μιας συνάρτησης που δέχεται ένα πίνακα αριθμών κινητής υποδιαστολής απλής ακρίβειας και επιστρέφει την τάξη του στοιχείου του που έχει την μεγαλύτερη τιμή.

### Απάντηση

```
int findMaxIndex(float arr[], int size) {  
    int maxIndex = 0;  
    for (int i = 1; i < size; i++) {  
        if (arr[i] > arr[maxIndex]) {  
            maxIndex = i;  
        }  
    }  
    return maxIndex;  
}
```

Τι κάνει ο υπολογιστής όταν εκτελεί την πρόταση `int fbn[24];`

**Απάντηση** Βρίσκει το στοιχείο του πίνακα `fbn` που βρίσκεται στην θέση 24 και το διαβάζει.

Διάρκεια 7 λεπτά  
2 Απαντήσεις και αυτές λάθος

```
int max(int nums[]) {  
    int i = 0;  
    int max = 0;  
    for (i=0; i < sizeof(nums)/sizeof(nums[0]); {  
        if (nums[i] > max) {  
            max=nums[i];  
        }  
    }  
}
```

[Average] Τι ρόλο παίζει η τοποθέτηση του `(float)` πριν από την μεταβλητή `sum` στην έκφραση υπολογισμού της τιμής της μεταβλητής που αναπαριστά τον μέσο όρο;

### Απάντηση

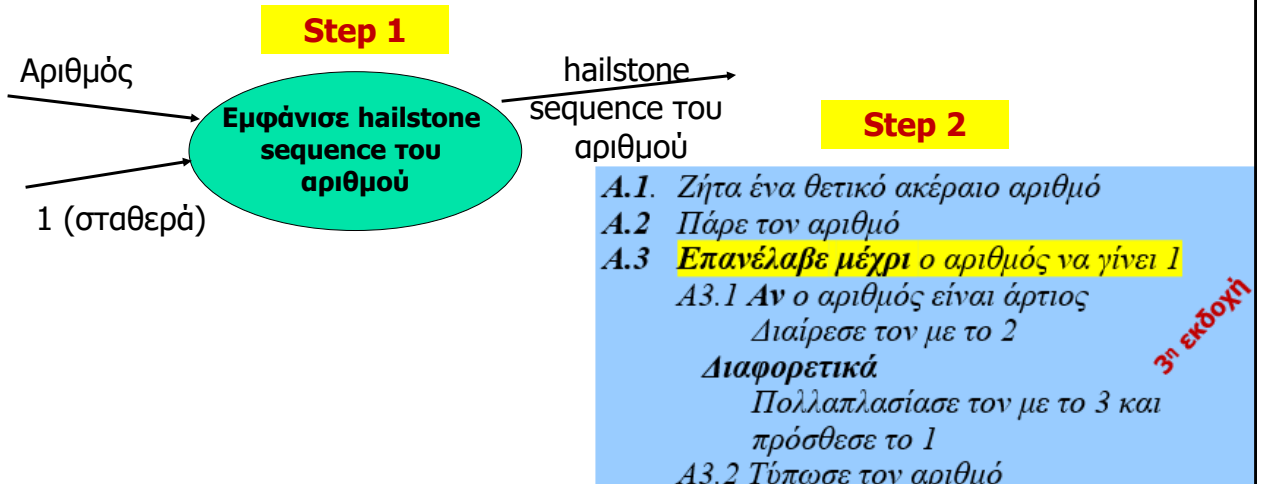
Σε περίπτωση που έχουν δοθεί `float` αριθμοί για την πράξη το αποτέλεσμα θα διαφέρει καθώς αυτά τα ψηφία μετά το κομμα θα ήταν υπο χρήσης `int 0`

### Απάντηση

Με την τοποθέτηση `float` πριν την μεταβλητή `sum` επιτρέπουμε στην `sum` να πάρει τιμές ακεραίων και δεκαδικών.

## HailstoneSequence – [S1, S2] Αφαιρετική αναπαράσταση

Αναπτύξτε ένα πρόγραμμα σύμφωνα με το οποίο το σύστημα θα δέχεται έναν αριθμό και θα εμφανίζει την hailstone sequence για τον αριθμό αυτό.





# HailstoneSequence – [S3]

1  
Πάρε τον αριθμό  
Επανάλαβε όσο ο αριθμός είναι διάφορος του 1  
Αν ο αριθμός είναι άρτιος  
    διαίρεσε με το 2.  
Αλλιώς  
    πολλαπλασίασε με το 3 και πρόσθεσε το 1  
Τύπωσε τον αριθμό

```
while(num != 1){  
    if(num%2==0)  
        num=num/2;  
    else {  
        num = num*3;  
        num = num+1;  
    }  
    printf("%d\n", num);  
}
```

2  
Πάρε τον αριθμό  
Επανάλαβε όσο ο αριθμός είναι διάφορος του 1  
Υπολόγησε επόμενο όρο ακολουθίας  
Εμφάνισε τον όρο

```
while(num!=1){  
    num=HSNextTerm(num);  
    printf("%d\n",num);  
}
```

3  
Πάρε τον αριθμό  
Υπολόγησε και Εμφάνισε Hailstone ακολουθία

```
calcAndDisplayHailstoneSequence(num);
```



Αναπτύξτε μια 4<sup>η</sup> έκδοση με βάση τις συναρτήσεις *calcHailstoneSequence* και *displayHailstoneSequence*

## Week 4 - Week 4 Αντικείμενο

- **Είσοδος – Έξοδος** ([Άσκηση 1 Κεφάλαιο 11](#))
  - Συναρτήσεις Εισόδου/Εξόδου της βασικής βιβλιοθήκης ([C stdio Functions stdio.h](#))
- **Ο τύπος του χαρακτήρα (char)** ([I2P Variables&DataTypes.pdf](#) s12,13,17-21)
  - American Standard Code for Information Interchange (ASCII).
- **Αλφαριθμητικά (String)** ([Άσκηση 2 κεφάλαιο 11](#)) ([I2P Variables&DataTypes.pdf](#), s23)
  - Το αλφαριθμητικό ως πίνακας ([I2P Arrays.pdf](#)) χαρακτήρων
  - Συναρτήσεις διαχείρισης αλφαριθμητικών της βασικής βιβλιοθήκης ([C string Functions string.h](#))
  - Χαρακτηριστικές συναρτήσεις: `strlen()`, `strcat()`, `strcmp()`, `strcpy()`, `strncat()`, `strncpy()`, ...
- **Δείκτες (Pointers)** ([Άσκηση 2 κεφάλαιο 11](#)) ([I2P Pointers.pdf](#))
  - Το όνομα του πίνακα ως διεύθυνση του πρώτου στοιχείου του πίνακα

## Week 5 - Οργάνωση

### ■ Παραδοτέο 4<sup>ης</sup> εβδομάδας (MinAndMax)

- Σχολιασμός Παραδοτέου
- Εναλλακτικές υλοποιήσεις MinAndMax (modularity)

### ■ String Handlings (addressBook case study)

- getString based on getchar()
- (Άσκηση 4 – Κεφάλαιο 11)

### ■ Ασκήσεις 5<sup>ης</sup> Εβδομάδας

- Άσκηση 3 (Κεφάλαιο 11) – Τέσσερις Πράξεις
- Άσκηση 4 (Κεφάλαιο 11) - Ταξινόμηση Λέξεων

## Παραδοτέο 4<sup>ης</sup> Εβδομάδας

Εργαστήριο 4ης εβδομάδας (Παραδοτέο)

Παρασκευή, 25 Οκτωβρίου 2024 - 10:30 π.μ. - από τον χρήστη Θραμπουλίδης Κλεάνθης

Το παραδοτέο της 4<sup>ης</sup> εβδομάδας αποτελείται από μια Εργασία η οποία περιλαμβάνει 2 αρχεία pdf:

1. Το ένα θα περιέχει τον πηγαίο κώδικα των εκδόσεων V1 και V2 που έχετε αναπτύξει και τρέχουν σωστά για την Δράση [Δ3] - MinAndMax (αν έχετε αναπτύξει μια στέλνετε μόνο μία). Το pdf που θα παραδώσετε θα το δημιουργήσετε από ένα αρχείο πηγαίου κώδικα στο οποίο θα κάνετε copy-paste τους πηγαίους κώδικες από τα αντίστοιχα αρχεία πηγαίου κώδικα των δύο εκδόσεων. Πριν από τον πηγαίο κώδικα κάθε έκδοσης θα βάλετε σε σχόλιο την έκδοση του, π.χ. // V1, //V2
2. Το άλλο θα περιέχει:

α) την αφαιρετική αναπαράσταση του προγράμματος, και,

β) τις λεκτικές περιγραφές των 2 εκδόσεων που έχετε περιλάβει στο αρχείο πηγαίου κώδικα με την ένδειξη της έκδοσης πριν από κάθε περιγραφή. Οι λεκτικές περιγραφές θα πρέπει να είναι σύμφωνες με τον πηγαίο κώδικα σας.

👍 2 🗨️ 0

Εργαστήριο 4ης εβδομάδας (Αντικείμενο) [Updated]

👍 3 🗨️ 0

## Παράδειγμα Παραδοτέου



### A) αφαιρετική παράσταση του προγράμματος V1:

Στόχος του προγράμματος είναι να διαβάσει ένα τιμές από τον χρήστη και να υπολογίζει τη μικρότερη και την μεγαλύτερη τιμή.

1. Ορίζω τις μεταβλητές min και max στις οποίες αποθηκεύονται οι άκαιρες τιμές και την μεταβλητή numberOfTerms που ορίζει το πλήθος των αριθμών που θα εισάγει ο χρήστης.
2. Ο χρήστης καλείται να δώσει το πλήθος των τιμών και στην συνέχεια τους αριθμούς έναν.
3. Για κάθε τιμή που εισάγεται αν είναι μεγαλύτερη από την τρέχουσα μεγαλύτερη ενημερώνεται η Max με την νέα τιμή, όμοιος αν η τιμή που εισάγεται είναι μικρότερη από την τρέχουσα μικρότερη ενημερώνεται η min.
4. Τέλος, εμφανίζονται οι τελικές τιμές min και max.

“έκανα την υποβολή των 2 pdf .... και μ ήρθε μήνυμα στο ακαδημαϊκό email .... ότι ο βαθμός είναι 0. Είναι ο βαθμός μου στο παραδοτέο ή κάποιο λάθος του συστήματος γιατί **έκανα ακριβώς ότι ζητούσαν οι οδηγίες.** Σας ευχαριστώ για τον χρόνο σας.”



### B) Λεκτική περιγραφή του προγράμματος V1:

Μετά την υποβολή σας εμφανίζεται βαθμός 0. Τι θα κάνετε;

1. Το πρόγραμμα ζητείται από τον χρήστη το πλήθος των αριθμών π θα βάλει.
2. ο χρήστης εισάγει τις τιμές μια.
3. το πρόγραμμα έχει μια τιμή min και max.
4. για κάθε αριθμό που εισάγει ο χρήστης ενημερώνεται η min αν η τιμή είναι μικρότερη από την min και η τιμή max αν η νέα τιμή είναι μεγαλύτερη από την max.
5. Τέλος εμφανίζει το μικρότερο και τον μεγαλύτερο αριθμό.

## Week 6 - Οργάνωση

### ■ Παραδοτέο 5<sup>ης</sup> εβδομάδας

- Σχολιασμός Παραδοτέου
- Προετοιμασία EA-5

### ■ String Handlings (addressBook case study)

- getString based on getchar() (getStringV2())
- Βιβλιοθήκη Τρίτου κατασκευαστή (third party library) – **i2p library**

### ■ Menu and Words Handling

- Δόμηση πηγαίου κώδικα προγράμματος
- Αξιοποίηση Αφαιρετικότητας – Αυξητική Ανάπτυξη

# Παραδοτέο 5ης Εβδομάδας

[EA-5] Εργαστήριο 5ης και 6ης εβδομάδας (Αντικείμενο)

[EA-5] Εργαστήριο 5ης και 6ης εβδομάδας (Αντικείμενο)

- [Δ1] – **StringHandling**  
(Διαχείριση Αλφαριθμητικού)
- [Δ2] – **MinAndMax**  
(Μέγιστο και ελάχιστο σε Ακολουθία αριθμών)
- [Δ3] – **Τέσσερις Πράξεις** (Menu)
- [Δ4] – **Διαχείριση Λέξεων** (WordsHandling)
- Η βιβλιοθήκη i2p

προϋποθέτει την προσεκτική παρακολούθηση διά λέξης και Φροντιστηρίου και μελέτη του αντίστοιχου υλικού στο eclass).

Παραδοτέο Εργαστηρίου 5ης Εβδομάδας  
Ανάθεση σε: Συγκεκριμένες Ομάδες Χρηστών

Από: Παρασκευή, 1 Νοεμβρίου 2024 - 6:30 μ.μ.  
έως: χθες - 11:59 μ.μ.

41

Κλεάνθης Θραμπουλίδης

Week Agenda

## βαθμός δυσκολίας EA-4

[Δ1] – BasicInputOutput (Βασική Είσοδος/Εξοδος)

[Δ2] – StringHandling (Διαχείριση Αλφαριθμητικού)

[Δ3] – MinAndMax (Μέγιστο και ελάχιστο σε Ακολουθία αριθμών)

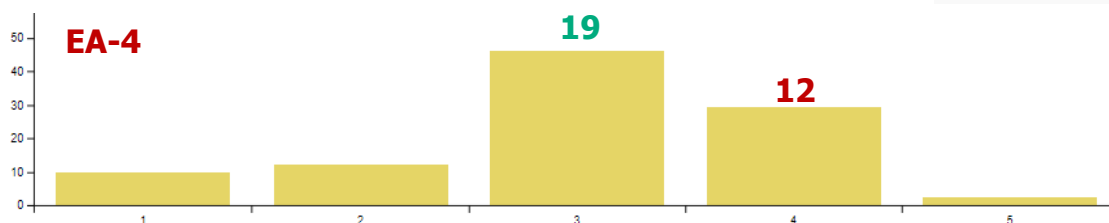
1- Πολύ εύκολη

2 - Εύκολη

3 - Βατή

4 - Δύσκολη

5 - Πολύ δύσκολη

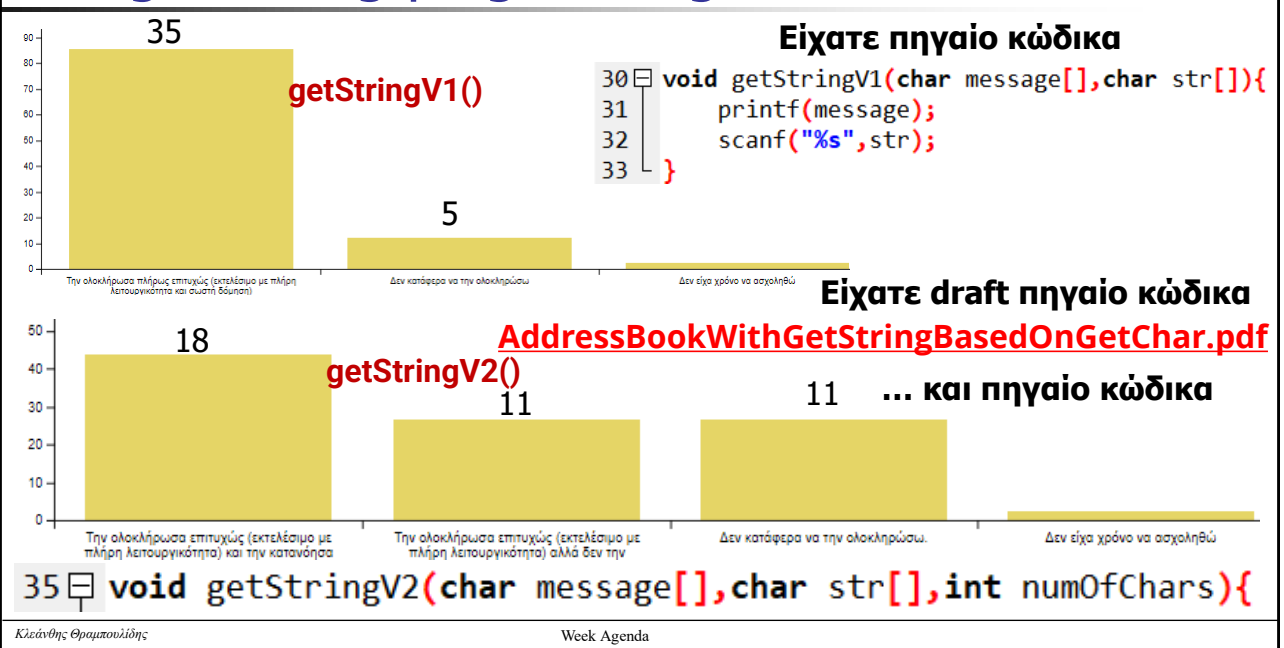


<https://eclass.upatras.gr/modules/blog/index.php?course=CEID1456&action=showPost&pId=1716>

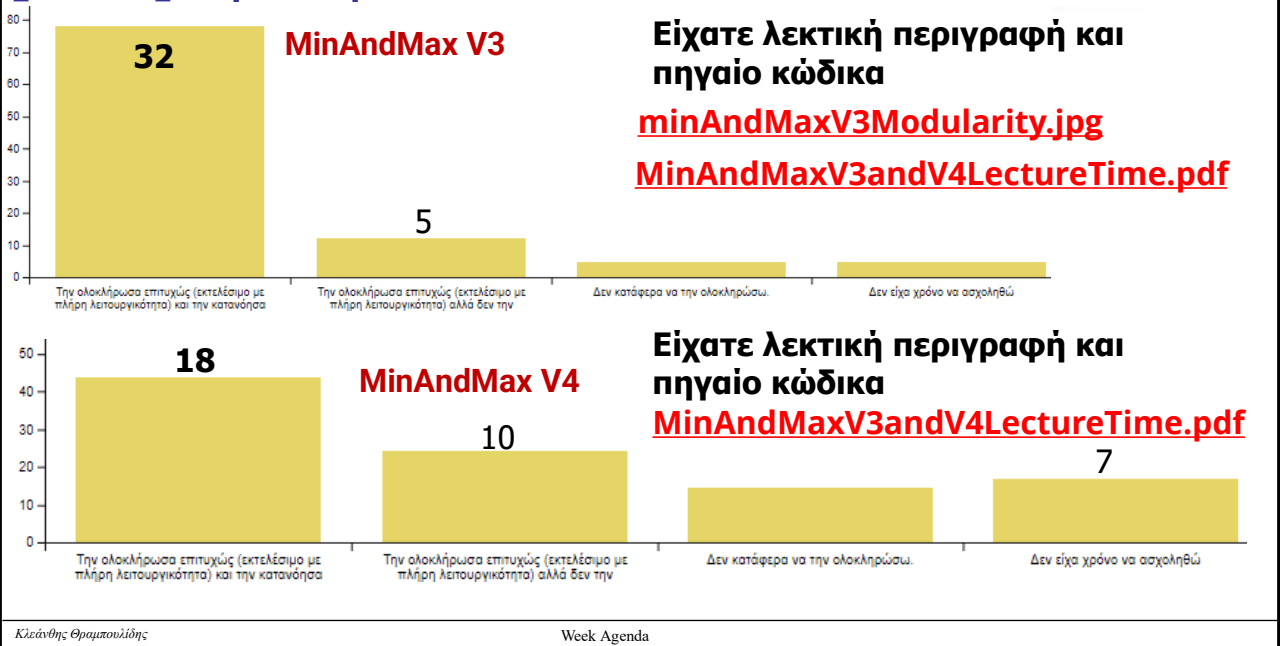
Κλεάνθης Θραμπουλίδης

Week Agenda

# StringHandling με getStringV1 και V2



# [EA-5] Δράση Δ2 – MinAndMax V3 and V4



## βαθμός δυσκολίας EA-5

1 - Πολύ εύκολη

2 - Εύκολη

3 - Βατή

4 - Δύσκολη

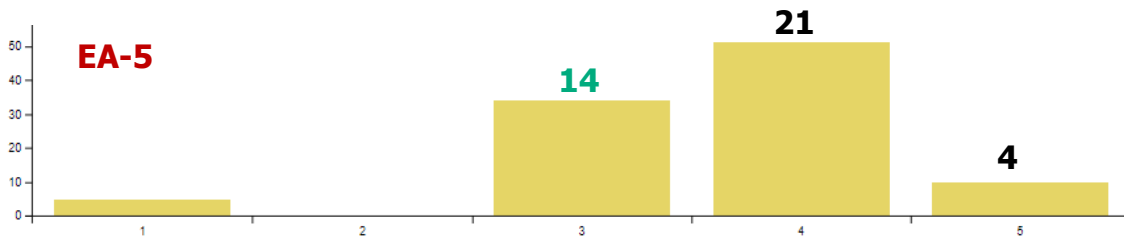
5 - Πολύ δύσκολη

[Δ1] – **StringHandling** (Διαχείριση Αλφαριθμητικού)

[Δ2] – **MinAndMax** (Μέγιστο και ελάχιστο σε Ακολουθία αριθμών)

[Δ3] – **Τέσσερις Πράξεις (Menu)**

[Δ4] – **Διαχείριση Λέξεων (WordsHandling)**



<https://eclass.upatras.gr/modules/blog/index.php?course=CEID1456&action=showPost&pId=1726>

Κλεάνθης Θραμπουλίδης

Week Agenda

## [EA-5] Δράση Δ3 – Menu



Δώσε λεκτική περιγραφή

### Λεκτική Περιγραφή

Εμφάνισε μενού (λειτουργίες προγράμματος)

Πάρε επιλογή χρήστη

Όσο η επιλογή δεν είναι τερματισμός προγράμματος

Εκτέλεσε τη ζητούμενη λειτουργία

Εμφάνισε μενού και πάρε την επιλογή χρήστη

Κλεάνθης Θραμπουλίδης

Week Agenda

# Βιβλιοθήκη I2P

## Οδηγίες χρήσης της βιβλιοθήκης I2P

Οδηγίες χρήσης της της βιβλιοθήκης I2P θα βρείτε στο [i2pLibraryV4part.pdf](#)

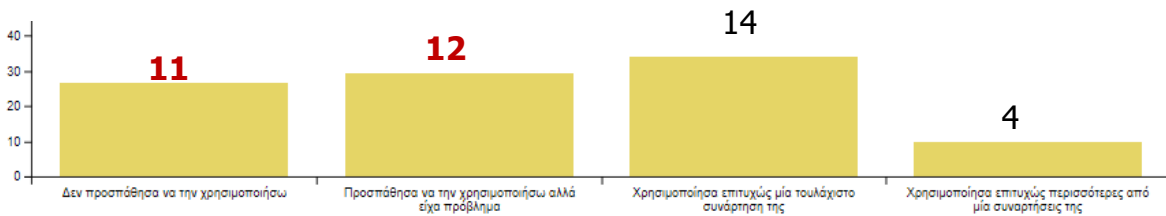
## Είχατε οδηγίες Χρήσης

[i2pLibraryV4part.pdf](#)

Μπορείτε να χρησιμοποιήσετε και τις συναρτήσεις `getInt()` και `getArrayOfInts()` των οποίων οι δηλώσεις είναι οι παρακάτω:

```
int getArrayOfInts(int ar[],int maxitems);
```

```
int getInt(char message[]);
```

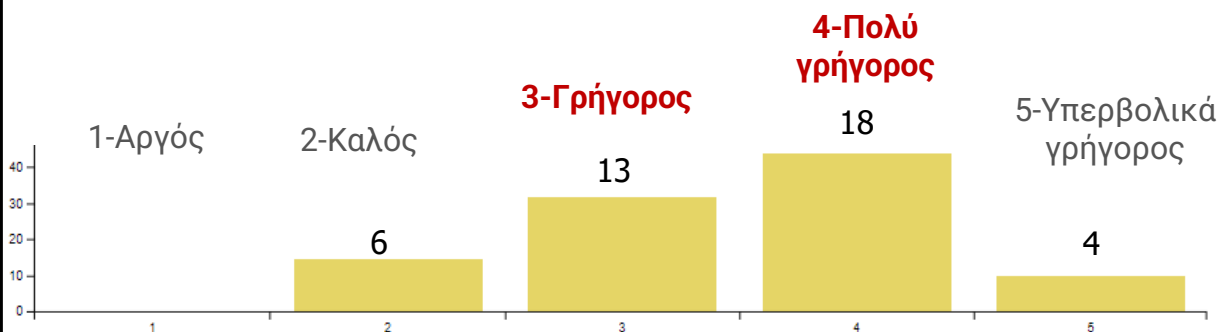


Κλεάνθης Θραμπουλίδης

Week Agenda

# ρυθμός μαθήματος συνολικά

Σημειώστε πως χαρακτηρίζετε τον ρυθμό με τον οποίο προχωράμε στο μάθημα συνολικά 1-Αργό, 2-Καλό, 3-Γρήγορο, 4-Πολύ γρήγορο, 5-Υπερβολικά γρήγορο



Κλεάνθης Θραμπουλίδης

Week Agenda

# Λεκτική περιγραφή της getStringV2

1.Ενημέρωσε χρήστη να δώσει όνομα 2.Πάρε όνομα 3.Ενημέρωσε χρήστη να δώσει επίθετο 4.Πάρε επίθετο 5.Ενημέρωσε χρήστη να δώσει οδό 6.Πάρε οδό 7.Ενημέρωσε χρήστη να δώσει νούμερο 8.Πάρε νούμερο 9.Εμφάνισε όλα τα στοιχεία στη σειρά



εκτύπωσε το μήνυμα που ζητάει τον αριθμό των χαρακτήρων δώσε τον αριθμό των χαρακτήρων διάβασε τον πρώτο χαρακτήρα αν ο πρώτος χαρακτήρας είναι το κενό τότε βάλε στην θέση 0 ότι τερματίζει η συνάρτηση μέχρι να φτάσεις τον αριθμό των χαρακτήρων-1 ή να βρεις το κενό, θα παίρνεις τον εκάστοτε χαρακτήρα και θα τον αποθήκευσε στην αντίστοιχη θέση βάλε στην επόμενη θέση τον τερματικό χαρακτήρα



ΕΜΦΑΝΙΣΕ ΜΗΝΥΜΑ ΓΙΑ ΟΣΟ ΕΙΝΑΙ Ο ΑΡΙΘΜΟΣ ΤΩΝ ΧΑΡΑΚΤΗΡΩΝ ΕΠΑΝΕΛΑΒΕ ΠΑΡΕ ΧΑΡΑΚΤΗΡΑ ΟΡΙΣΕ ΤΟ ΕΚΑΣΤΟΤΕ ΣΤΟΙΧΕΙΟ ΤΟΥ ΑΛΦΑΡΙΘΜΗΤΙΚΟΥ ΩΣ ΤΟΝ ΧΑΡΑΚΤΗΡΑ ΠΟΥ ΔΑΙΒΑΣΕΣ ΑΝ Ο ΧΑΡΑΚΤΗΡΑΣ ΕΙΝΑΙ ΤΟ ΕΝΤΕΡ ΕΠΕΣΤΡΕΨΕ ΟΡΙΣΕ ΤΟ ΕΠΟΜΕΝΟ ΣΤΟΙΧΕΙΟΥ ΤΟΥ ΑΛΦΑΡΙΘΜΗΤΙΚΟΥ ΙΣΟ ΜΕ ΤΟ 0

Κυριε Θραμπουλιδη, δεν μπορεσα να ερθω στο εργαστηριο της 5ης βδομαδας, ομως τα προσπαθηα στο σπιτι μου και σας στελνω feedback απο το σπιτι μου.

Διάβασε αλφαριθμητικό Διάβασε μόνο όσους χαρακτήρες μπορεί να αποθηκεύσει η αντίστοιχη μεταβλητή

Ζήτα λέξη από τον χρήστη Σε περίπτωση λάθους εμφάνισε του μήνυμα κινδύνου



1) Τύπωσε το μήνυμα 2) Μέχρι να συμπληρωθεί ο μέγιστος αριθμός χαρακτήρων ή μέχρι να φτάσει χαρακτήρα "\n" επανέλαβε: 2.1) Διάβασε χαρακτήρα 2.2) Αποθήκευσε χαρακτήρα στον πίνακα 2.3) Α χαρακτήρας είναι "\n", ολοκλήρωσε το αλφαριθμητικό 3) Ολοκλήρωσε το αλφαριθμητικό

Παρε τη συμβολοσειρα με οριο χαρακτηρων Αποθηκευσε την με ασφαλη τερματισμο

Τύπωσε ένα μήνυμα. Πάρε χαρακτήρες.



Δώσε οδό Δώσε όνομα Δώσε επίθετο Για κάθε λέξη μικρότερη του μέγιστου αριθμού χαρακτήρων Πάρε λέξη Αποθήκευσε την λέξη



Ζήτα όνομαεπίθετοοδό Αν είναι κενό τερμάτισε Επανάλαβε ζήτα όνομαεπίθετοοδό αποθήκευσε την πληροφορία(όνομα|ειδιεύθυνση|οοδ) διαδοχικά σε μικρά κομμάτια Τερμάτισε

Πάρε 4 χαρακτήρες για την οδό Πάρε 3 χαρακτήρες για το επώνυμο Πάρε 5 χαρακτήρες για το όνομα



Η συνάρτηση getStringV2 λαμβάνει από τον χρήστη μια σειρά αλφαριθμητών τα οποία διαβάζει ξεχωριστά το καθένα. Ελέγχει τον αριθμό των αλφαριθμητών που δόθηκαν από τον χρήστη σε σχέση με το επιτρεπόμενο αριθμό από το πρόγραμμα. Εξάγει στον χρήστη ή την σειρά αλφαριθμητών ή μήνυμα που τον ενημερώνει για την παραβίαση του μέγιστου αριθμού αλφαριθμητών εάν έχει παραβιαστεί.

1. Εκτύπωσε μήνυμα 2. Διάβασε τον 1ο χαρακτήρα 3. Έλεγξε αν είναι κενό 4. Διάβασε χαρακτήρες και αποθήκευσε τους σε πίνακα 5. Πρόσθεσε /0 στο τέλος του πίνακα

1.Εκτύπωσε μήνυμα 2.Διαβάσε 1ο χαρακτήρα 3.Αν enter τερμάτισε Άλλως Μεχρι να τελειωσουν οι χαρακτες ή να συναντησεις enter -Διαβαζε καθε χαρακτηρα -Αποθηκευσε τον σε θεση του string



Κλεάνθης Θραμπουλιδής

Week Agenda

# Λεκτική περιγραφή της getStringV2

## Κώδικας που είχαμε δώσει πρόχειρα

```
34 // 1st darft version - to be updated
35 void getStringV2(char message[],char str
36     char ch;
37     int i=0;
38
39     printf(message);
40     ch=getchar();
41     if(ch=='\n'){
42         str[0]='\0';
43         return;
44     }
45     for(i=0;i<numOfChars&&ch!='\n';i++){
46         ch=getchar();
47         str[i]=getchar();
48     }
49     str[i]='\0';
50     return;
51 }
```

## Λεκτική Περιγραφή

Εμφάνισε μήνυμα

Για όσους χαρακτήρες ζητείται

Πάρε χαρακτήρα

Αν ο χαρακτήρας είναι Enter

τερμάτισε την επανάληψη

Βάλε τον χαρακτήρα στο αλφαριθμητικό

Βάλε τον χαρακτήρα τερματισμού μετά τον τελευταίο χαρακτήρα

*Καθάρισε βασική είσοδο*

Μετατρέψτε την Λεκτική Περιγραφή σε πηγαίο κώδικα

Κλεάνθης Θραμπουλιδής

Week Agenda



# Words Handling

word1, word2, ... word5 είναι πέντε μεταβλητές ίδιου τύπου

```
char word1[20];  
char word2[20];  
char word3[20];  
char word4[20];  
char word5[20];
```



**char word[5][20];**



word[0] ↔ word1  
word[1] ↔ word2  
word[2] ↔ word3  
word[3] ↔ word4  
word[4] ↔ word5



```
scanf("%s",word1);  
printf("%s\n", word1);
```

```
scanf("%s",word[0]);  
printf("%s\n", word[0]);
```

```
scanf("%s",word[i]);  
printf("%s\n", word[i]);
```



# Βασικά Στοιχεία της C

**Το αντικείμενο του μαθήματος δεν είναι η C.** Αποφύγετε στην φάση αυτή λεπτομέρειες της C! Ότι χρειάζεται από την C για το μάθημα Εισαγωγή στον Προγραμματισμό θα βρείτε στους παραπλεύρως συνδέσμους από την κατηγορία συνδέσμους από την κατηγορία Βασικά στοιχεία της Γλώσσας C.

Ομάδες Χρηστών

Συζητήσεις

Σύνδεσμοι

Συνομιλία

Κατηγορίες συνδέσμων

Βασικά στοιχεία της Γλώσσας C

Γλώσσα Προγραμματισμού C

Κατηγορίες συνδέσμων

Βασικά στοιχεία της Γλώσσας C

I2P\_LanguageOverview.pdf

Συναρτήσεις (I2P\_Functions.pdf)

Μεταβλητές και Τύποι Δεδομένων (I2P\_Variables&DataTypes.pdf)

Τελεστές και Εκφράσεις (I2P\_Expressions&Operators.pdf)

Προτάσεις Ελέγχου Ροής (I2P\_ControlFlowStatements.pdf)

Πίνακες (I2P\_Arrays.pdf)

Δείκτες (I2P\_Pointers.pdf)

I2P\_StatementsInBrief.pdf

Δομές (I2P\_Struct.pdf)

<https://eclass.upatras.gr/modules/link/?course=CEID1456>

## Week 7 - Οργάνωση

[EA-6] Εργαστήριο 7ης εβδομάδας (Αντικείμενο)

- **[EA-5] Σχολιασμός**
- **Παραδοτέο 6ης εβδομάδας**
  - Σχολιασμός Παραδοτέου
- **Προετοιμασία EA-6**
  - **MostFrequentNumber**
  - Divisors
  - **AddNumbers**

### Δράσεις

Η EA-5 αποτελείται από 3 δράσεις που δίνονται παρακάτω.

Σε όλες τις δράσεις θα ακολουθήσετε τα βήματα της Διαδικασίας Ανάπτυξης Προγράμματος (Updated) αξιοποιώντας ότι υλικό ε abstraction) και αυξητική ανάπτυξη (incremental development)

### [Δ1] - MostFrequentNumber

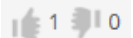
Αναπτύξτε ένα πρόγραμμα σύμφωνα με το οποίο το σύστημα εμφανίστηκε στην ακολουθία περισσότερες συνεχόμενες φορές

### [Δ2] - Divisors

Αναπτύξτε ένα πρόγραμμα σύμφωνα με το οποίο το σύστημα ε είναι πρώτος θα εμφανίζει τους διαιρέτες του και το πλήθος τ

### [Δ3] - AddNumbers

Αναπτύξτε ένα πρόγραμμα με βάση το οποίο η μηχανή θα εκτε **I2P\_BasicConceptsPartC.pdf**.



<https://eclass.upatras.gr/modules/blog/index.php?course=CEID1456&action=showPost&pId=1735>

## Week 7 - EA-5 Επιλεγμένα θέματα

- **[Δ1] StringHandling**
  - Μετατροπή λεκτικής περιγραφής σε πηγαίο κώδικα
- **[Δ2] MinAndMax**
  - Χρήση της i2p library (sortIntArrayInc) [Η Βιβλιοθήκη i2p](#)
- **[Δ3] MenuV3** [menuV3.pdf](#)
  - Μετατρέψτε τις συναρτήσεις add(), sub(), mul() κ.λ.π. σε επαναχρησιμοποιήσιμες. [Συναρτήσεις \(I2P\\_Functions.pdf\)](#) [Βασικά στοιχεία της Γλώσσας C](#)
- **[Δ4] WordsHandling** [wordsHandlingV5.pdf](#)
  - Λεκτική περιγραφή των συναρτήσεων wordStats() και charStats()

# Παραδοτέο 6ης Εβδομάδας

[EA-5] Εργαστήριο 5ης και 6ης εβδομάδας (Αντικείμενο)

## ■ Αναφέρεται στην Δράση [Δ4] (WordsHandling) της [EA-5]

Στο **Μέρος 1°** θα παραδώσετε ένα αρχείο, με όνομα <Ζώνη-Θέση>\_SourceCode.pdf.

Το αρχείο θα περιέχει τον **πηγαίο κώδικα της τελευταίας λειτουργούσας έκδοσης** που αναπτύξατε για τη Δράση [Δ4] - WordsHandling.



Στο **Μέρος 2°** θα παραδώσετε ένα αρχείο, με όνομα <Ζώνη-Θέση>\_VD.pdf, το οποίο θα περιέχει τις λεκτικές περιγραφές για τα παρακάτω:


1. Λεκτική περιγραφή της main() της έκδοσης της wordsHandling της οποίας παραδώσατε τον πηγαίο κώδικα.
2. Λεκτική περιγραφή της charStats(), ή όπως αλλιώς έχετε ονομάσει την συνάρτηση που υλοποιεί την λειτουργία στατιστικών χαρακτήρα). Η λεκτική περιγραφή θα πρέπει να είναι σύμφωνη με τον πηγαίο

προϋποθέτει την προσεκτική παρακολούθηση διά λέξης και φροντιστηρίου και μελέτη του αντίστοιχου υλικού στο eclass.


Παραδοτέο Εργαστηρίου 6ης εβδομάδας (Μέρος 1ο) Ατομική εργασία Ανάθεση σε: Συγκεκριμένες Ομάδες Χρηστών	TurnItIn	72
Παραδοτέο Εργαστηρίου 6ης εβδομάδας (Μέρος 2ο) Ατομική εργασία	TurnItIn	67

## Ενδεικτικά παραδοτέα – main()

- Παρουσίασε μενού και πάρε επιλογή από χρήστη 
- Ανάλογα την επιλογή του χρήστη κάνε αντίστοιχη δράση 
- Ζήτα πάλι την επιλογή του χρήστη

1. ΕΜΦΑΝΙΣΕ ΣΤΟΝ ΧΡΗΣΤΗ ΤΗΣ ΕΠΙΛΟΓΗΣ ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ
2. ΠΑΡΕ ΑΡΙΘΜΟ ΑΠΟ ΤΟΝ ΧΡΗΣΤΗ ΓΙΑ ΤΗΝ ΕΠΙΛΟΓΗ ΠΡΟΓΡΑΜΜΑΤΟΣ
3. ΑΝΑΖΗΤΗΣΕ ΜΕ ΒΑΣΗ ΤΟΝ ΑΡΙΘΜΟ ΤΟ ΠΡΟΓΡΑΜΜΑ
4. ΑΝ ΔΕΝ ΥΠΑΡΧΕΙ ΕΜΦΑΝΙΣΕ ΑΝΑΛΟΓΟ ΜΗΝΥΜΑ 
5. ΕΠΑΝΕΛΑΒΕ ΜΕΧΡΙ ΤΟΝ ΤΕΡΜΑΤΙΣΜΟ ΑΠΟ ΤΟΝ ΧΡΗΣΤΗ
6. ΣΤΟΝ ΤΕΡΜΑΤΙΣΜΟ ΕΜΦΑΝΙΣΕ ΑΝΑΛΟΓΟ ΜΗΝΥΜΑ

Περιγραφή της main()

-Το πρόγραμμα ξεκινά με την εκτύπωση ενός εισαγωγικού μηνύματος ( 


Λεκτική Περιγραφή main(): 

- 1) Όρισε μεταβλητή ch
- 2) Όρισε μεταβλητή max
- 3) Όρισε μεταβλητή min

Δείξε τον τίτλο προγράμματος

Όσο δεν έχει επιλεγθεί ο τερματισμός του προγράμματος:

Ενάλλασε μεταξύ των λειτουργιών:

Εισήγαγε λέξεις 

Τέλος


Πρόβαλε λέξεις


Τέλος


Ταξινόμησε λέξεις σε αύξουσα σειρά

Τέλος

A) Λεκτική Περιγραφή της main() 

Η συνάρτηση main() είναι η κύρια συνάρτηση τ την εκτέλεση του μενού και την επιλογή των δι της είναι η εξής: 

Εμφάνιση Μενού: 

Όρισε την ακαίρεα μεταβλητή Operation  
Τύπωσε "WordHandling"  
Τρέξε την συνάρτηση menu 

## Ενδεικτικά παραδοτέα – charStats()

Ζήτα χαρακτήρα από τον χρήστη

Έλεγε κάθε χαρακτήρα των λέξεων που ο χρήστης έχει προσθέσει αν είναι ίδιος με τον χαρακτήρα που έδωσε

Τύπωνε πόσες φορές εμφανίζεται ο χαρακτήρας συνολικά



Έλεξε σε κάθε λέξη ποσες φορές υπάρχει ο κάθε χαρακτήρας .



Τοποθέτησε τον κάθε αριθμό χαρακτήρων σε έναν πίνακα.

Τύπωνε τον κάθε αριθμό.



1. Θέσε μεταβλητές « j , biggest , smallest, NumSum ,avg»
2. Θέσε την τιμή biggest ως το μήκος της word[0]
3. Θέσε την τιμή smallest ως το μήκος της word[0]
4. Θέσε την NumSum ως μηδέν
5. Εμφάνισε «word statistics»



### ΛΕΚΤΙΚΗ ΠΕΡΙΓΡΑΦΗ ΤΗΣ WORDSSTATS

1. ΑΝ ΔΕΝ ΕΧΕΙ ΔΩΣΕΙ Ο ΧΡΗΣΤΗΣ ΛΕΞΕΙΣ ΕΜΦ/
2. ΟΡΙΣΕ ΩΣ ΜΕΓΙΣΤΟ,ΕΛΑΧΙΣΤΟ ΚΑΙ ΜΕΣΟ ΜΗΚ
3. ΑΝΑΖΗΤΗΣΕ ΣΤΑΤΙΣΤΙΚΑ ΤΟΥ ΜΗΚΟΥΣ ΤΗΣ ΛΙ
6. Επανάλαβε όσο το j είναι μικρότερο του i αυξάνοντας το κατά 1

Κλεάνθης Θραμπουζίδης

Week Agenda

## MostFrequentNumber

### Συχνότερα εμφανιζόμενος αριθμός

- Αναπτύξτε ένα πρόγραμμα σύμφωνα με το οποίο το σύστημα θα διαβάζει μία ακολουθία από n αριθμούς και θα εμφανίζει:
  - α) τον αριθμό που εμφανίστηκε στην ακολουθία περισσότερες συνεχόμενες φορές, και,
  - β) το πλήθος των συνεχόμενων εμφανίσεων του.

### Διαδικασία Ανάπτυξης Προγράμματος (Updated)

- **[Step1-(S1)]** Δώστε την **Αφαιρετική αναπαράσταση** της Διεργασίας. Η Αφαιρετική αναπαράσταση **καταγράφει τη διεργασία και α) τις εισόδους** που αυτή απαιτεί για την εκτέλεση του έργου **και β) τις εξόδους της**.
  - Τα βήματα της διεργασίας θα περιγράφει η λεκτική περιγραφή.
- **[S2]** **Δώστε τη λεκτική περιγραφή** της διεργασίας **αξιοποιώντας αφαιρετικότητα στις διεργασίες**.
  - **Ελέγξτε τη λεκτική περιγραφή**  
**Ελέγξτε αν** εσείς, εκτελώντας τις ενέργειες που η λεκτική περιγραφή ορίζει, πετυχαίνετε τον στόχο της διεργασίας, για παράδειγμα να δημιουργήσετε την ακολουθία Fibonacci.
- **[S3]** Μετασχηματίστε τη λεκτική περιγραφή σε **C κώδικα**. Δηλώστε τις μεταβλητές που αντιστοιχούν σε εισόδους και εξόδους. Προσδιορίστε ποιες από τις εισόδους είναι μεταβλητές και ποιες σταθερές. Συνεχίστε με τις προτάσεις της Λ.Π.
- **[S4]** Μετασχηματίστε τον πηγαίο κώδικα σε εκτελέσιμο.
- **[S5]** Εκτελέστε και ελέγξτε το πρόγραμμα.

Ο αναλυτικός έλεγχος μπορεί να αναβληθεί εν μέρη για το S5

[I2P\\_BasicConceptsPartC.pdf](#)

Κλεάνθης Θραμπουζίδης

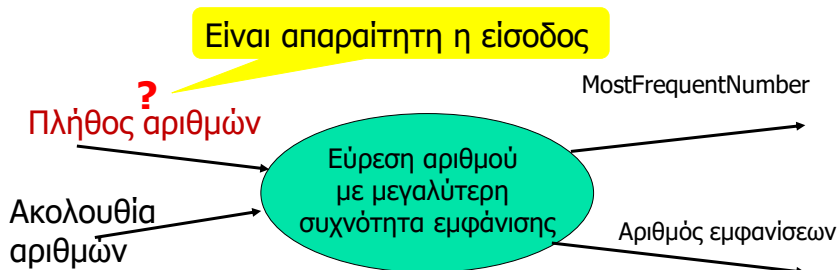
Week Agenda

40

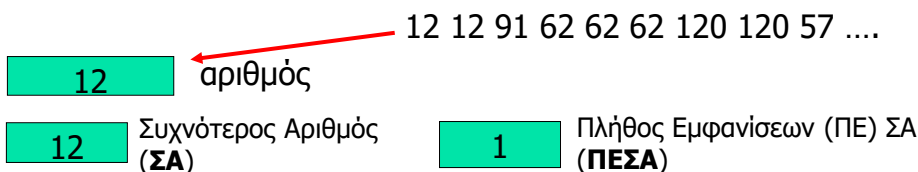
# MostFrequentNumber – S1 (Αφαιρετική Αναπαράσταση)

## Συχνότερα εμφανιζόμενος αριθμός

- Αναπτύξτε ένα πρόγραμμα σύμφωνα με το οποίο το σύστημα θα διαβάσει μία ακολουθία από  $n$  αριθμούς και θα εμφανίζει:
  - α) τον αριθμό που εμφανίστηκε στην ακολουθία περισσότερες συνεχόμενες φορές, και,
  - β) το πλήθος των συνεχόμενων εμφανίσεων του.



# MostFrequentNumber – S2 (Λεκτική Περιγραφή 1/3)



Πάρε τον πρώτο αριθμό  
Θεώρησε τον αριθμό ΣΑ με ΠΕ 1  
**Για κάθε** επόμενο αριθμό της ακολουθίας

- **Αν** αυτός είναι ίσος με τον ΣΑ
  - Αύξησε κατά 1 το ΠΕΣΑ
- **Αλλιώς**
  - Αντικατέστησε τον ΣΑ με τον αριθμό

Εμφάνισε Συχνότερο Αριθμό και Πλήθος Εμφανίσεων

1<sup>η</sup> εκδοχή

Αναγνωσιμότητα  
(Readability)  
Απλότητα  
(Simplicity)

Συνεχίστε τον έλεγχο  
για διάφορα σενάρια  
ακολουθίας

## MostFrequentNumber – S2 (Λεκτική Περιγραφή 2/3)

62 αριθμός

12 12 91 62 62 62 120 120 57 ....

62

Υποψήφιος Συχνότερος Αριθμός (ΥΣΑ)

3

Πλήθος Εμφανίσεων ΥΣΑ (ΠΕΥΣΑ)

12

Συχνότερος Αριθμός (ΣΑ)

2

Πλήθος Εμφανίσεων ΣΑ (ΠΕΣΑ)

Πάρε τον πρώτο αριθμό

Θεώρησε τον αριθμό ΥΣΑ με ΠΕ 1

Για κάθε επόμενο αριθμό της ακολουθίας

■ **Αν** αυτός είναι ίσος με τον ΥΣΑ

■ Αύξησε κατά 1 το ΠΕΥΣΑ

■ **Αν** ο ΠΕΥΣΑ είναι μεγαλύτερος από το ΠΕΣΑ

■ Αντικατέστησε τον ΣΑ με τον ΥΣΑ

■ Αντικατέστησε το ΠΕΣΑ με ΠΕΥΣΑ

■ **Αλλιώς**

■ Αντικατέστησε τον ΥΣΑ με τον αριθμό

Εμφάνισε Συχνότερο Αριθμό και Πλήθος Εμφανίσεων

2<sup>η</sup> εκδοχή

Συνεχίστε τον έλεγχο για διάφορα σενάρια ακολουθίας

## MostFrequentNumber – S2 (Λεκτική Περιγραφή 3/3)

12 12 91 62 62 62 120 120 57 ....

Πάρε τον πρώτο αριθμό

Θεώρησε τον αριθμό ΥΣΑ με ΠΕ 1

Για κάθε αριθμό της ακολουθίας

■ **Αν** αυτός είναι ίσος με τον ΥΣΑ

■ Αύξησε κατά 1 το ΠΕΥΣΑ

■ **Αν** ο ΠΕΥΣΑ είναι μεγαλύτερος από το ΠΕΣΑ

■ Αντικατέστησε τον ΣΑ με τον ΥΣΑ

■ Αντικατέστησε το ΠΕΣΑ με ΠΕΥΣΑ

■ **Αλλιώς**

■ Αντικατέστησε τον ΥΣΑ με τον αριθμό

Εμφάνισε ΣΑ και ΠΕΣΑ

Αναγνωσιμότητα  
(Readability)  
Απλότητα  
(Simplicity)

Συνεχίστε τον έλεγχο για διάφορα σενάρια ακολουθίας.

## Divisors

### Διαιρέτες αριθμού

- Αναπτύξτε ένα πρόγραμμα σύμφωνα με το οποίο το σύστημα θα δέχεται ως είσοδο ένα αριθμό και θα ελέγχει αν αυτός είναι πρώτος (prime). Αν ΔΕΝ είναι πρώτος θα εμφανίζει τους διαιρέτες του και το πλήθος τους.

### “Αναγνώριση πρώτου αριθμού και παραγοντοποίηση ακεραίου

Υπάρχουν ποικίλες μέθοδοι για να προσδιορίσουμε αν ένας αριθμός  $n$  είναι πρώτος. Η πιο βασική μέθοδος, η δοκιμαστική διαίρεση, έχει μικρή πρακτική χρησιμότητα επειδή είναι αργή. Ένα τμήμα των σύγχρονων μεθόδων για τον προσδιορισμό αν ένας αριθμός είναι πρώτος είναι εφαρμόσιμο για όλους τους αριθμούς, ενώ οι πιο αποτελεσματικές μέθοδοι είναι διαθέσιμες μόνο για συγκεκριμένες κατηγορίες αριθμών. Οι περισσότερες από αυτές τις μεθόδους λένε μόνο αν ο αριθμός είναι πρώτος ή όχι. Οι μέθοδοι, οι οποίες επιπλέον βρίσκουν και έναν ή περισσότερους παράγοντες του υπό εξέταση αριθμού ονομάζονται αλγόριθμοι παραγοντοποίησης.” wikipedia

### “Αλγόριθμοι εύρεσης πρώτων

Παρατίθενται μερικοί αλγόριθμοι (κατά σειρά ταχύτητας ή και απλότητας) για την εύρεση αν ο  $N \geq 2$  είναι πρώτος.

**Απλός 1** - Εξετάζουμε διαδοχικά όλους τους ακεραίους  $M < N$

**Απλός 2** - εξετάζοντας όλους τους αριθμούς  $M < N/2$

**Απλός 3** - εξετάζοντας όλους τους αριθμούς  $M$  που είναι μικρότεροι από την τετραγωνική ρίζα του  $N$ ,

**Απλός 4** - Εφαρμόζοντας το Θεώρημα του Ουίλσον

wikipedia

## Divisors - Λεκτική περιγραφή

### Διαιρέτες αριθμού

- Αναπτύξτε ένα πρόγραμμα σύμφωνα με το οποίο το σύστημα θα δέχεται ως είσοδο ένα αριθμό και θα ελέγχει αν αυτός είναι πρώτος (prime). Αν ΔΕΝ είναι πρώτος θα εμφανίζει τους διαιρέτες του και το πλήθος τους.

### Λεκτική Περιγραφή – Αφαιρετικότητα στις διεργασίες

- Πάρε τον αριθμό
- Αν ο **αριθμός είναι Πρώτος**
  - Ενημέρωσε τον χρήστη για αυτό
- Αλλιώς
  - **Εμφάνισε διαιρέτες του αριθμού**

Δώστε στην φάση αυτή μόνο την μετατροπή αυτής της ΛΠ σε πηγαίο κώδικα

# Divisors - Λεκτική περιγραφή -> SC

## Λεκτική Περιγραφή – Αφαιρετικότητα στις διεργασίες

Δες εναλλακτική υλοποίηση της getInt στο [I2P\\_Functions.pdf](#)

- Πάρε τον αριθμό
- Αν ο **αριθμός είναι Πρώτος**
  - Ενημέρωσε τον χρήστη για αυτό
- Αλλιώς
  - **Εμφάνισε διαιρέτες του αριθμού**

```
num=getInt("Dose arithmo:");  
if(isPrime(num))  
    printf("0 arithmos %d einai protos\n",num);  
else  
    displayDivisors(num);
```

```
1 #include <stdio.h>  
2 #include <stdlib.h>  
3 #include <stdbool.h>  
22 bool isPrime(int num){  
23     printf("isPrime executed for num =%d\n",num);  
24     // return true;  
25     return false;  
26 }
```

Ορίζει τον τύπο bool

[DivisorsV1.pdf](#)

# AddNumbers – Incremental Development

Γράψτε ένα πρόγραμμα σύμφωνα με το οποίο η μηχανή θα υπολογίζει το άθροισμα ακέραιων αριθμών. Προσέξτε τους περιορισμούς!

**Περιορισμοί:** Υποθέστε πως η μηχανή :

- δεν μπορεί να υπολογίσει άθροισμα αριθμών με περισσότερα από ένα ψηφία. Για τον υπολογισμό του αθροίσματος 2 ψηφίων χρησιμοποιήστε τον τελεστή +.
- μπορεί να κάνει τις πράξεις του πολλαπλασιασμού και διαίρεσης ακέραιων αριθμών με χρήση των τελεστών \* και / αντίστοιχα. Επιπλέον μπορεί να βρίσκει το modulo ενός ακέραιου αριθμού με χρήση του τελεστή %.

- 1η έκδοση: Οι αριθμοί είναι 2 με 2 το πολύ ψηφία.
- 2<sup>η</sup> έκδοση: Οι αριθμοί είναι 2 με 3 το πολύ ψηφία.
- 3<sup>η</sup> έκδοση: Οι αριθμοί είναι 3 με 3 το πολύ ψηφία.
- 4<sup>η</sup> έκδοση: Οι αριθμοί είναι k με n το πολύ ψηφία. **Αξιοποιήστε προτάσεις επανάληψης και πίνακες.**

Απαραίτητη η εφαρμογή της **Αυξητικής Ανάπτυξης**

[I2P\\_ControlFlowStatements.pdf](#)

[I2P\\_Arrays.pdf](#)



## AddNumbers – S1 (Αφαιρετική Αναπαράσταση)

### AddNumbers

Γράψτε ένα πρόγραμμα σύμφωνα με το οποίο η μηχανή θα υπολογίζει το άθροισμα ακέραιων αριθμών. Προσέξτε τους περιορισμούς!



$$\begin{array}{r} 82 \\ + 35 \\ \hline \end{array}$$

## AddNumbers – S2 (Λεκτική Περιγραφή 1)

$$\begin{array}{r} 82 \\ + 35 \\ \hline \end{array}$$

### Προϋποθέτει:

γνώση του αλγορίθμου και δυνατότητα περιγραφής του στην ομιλούμενη γλώσσα

### Είναι

η περιγραφή, σε προστακτική μορφή, των ενεργειών που πρέπει να κάνει ο **μαθητής δημοτικού** για να οδηγηθεί στο αποτέλεσμα με βάση τις εισόδους.

- Βάλτε τον ένα αριθμό κάτω από τον άλλο με στοίχιση στα δεξιά
- Πρόσθεσε το πρώτο δεξιά ψηφίο του 1<sup>ου</sup> αριθμού με το πρώτο δεξιά του 2<sup>ου</sup>
- Αν το άθροισμα είναι μικρότερο από 10
  - Γράψε το από κάτω
- Αλλιώς
  - Γράψε από κάτω το δεξιό ψηφίο του αθροίσματος
  - Κράτα ως κρατούμενο το αριστερό ψηφίο του αθροίσματος
- Πρόσθεσε το επόμενο ψηφίο του 1ου αριθμού με το επόμενο του 2ου
- Αν υπάρχει από προηγούμενη πρόσθεση κρατούμενο
  - πρόσθεσε το
- Γράψε το άθροισμα από κάτω

1<sup>η</sup> εκδόχη

Συνεχίστε τον έλεγχο για διάφορα σενάρια.

## AddNumbers – S2 (Λεκτική Περιγραφή 2)

$$\begin{array}{r} 82 \\ + 35 \\ \hline \end{array}$$

- Βάλε τον ένα αριθμό κάτω από τον άλλο με στοίχιση στα δεξιά
- Πρόσθεσε το πρώτο δεξιά ψηφίο του 1<sup>ου</sup> αριθμού με το πρώτο δεξιά του 2<sup>ου</sup>
- Αν το άθροισμα είναι μικρότερο από 10
  - Ενημέρωσε το τελικό άθροισμα με το ψηφίο του αθροίσματος
- Αλλιώς
  - Ενημέρωσε το τελικό άθροισμα με το τελευταίο δεξιά ψηφίο του αθροίσματος
  - Κράτα ως κρατούμενο το αριστερό ψηφίο του αθροίσματος
- Πρόσθεσε το επόμενο ψηφίο του 1ου αριθμού με το επόμενο του 2ου
- Αν υπάρχει από προηγούμενη πρόσθεση κρατούμενο
  - πρόσθεσε το
- Ενημέρωσε το τελικό άθροισμα με τα ψηφία του αθροίσματος
- Εμφάνισε το τελικό άθροισμα

2<sup>η</sup> εκδοχή

Συνεχίστε τον έλεγχο για διάφορα σενάρια.

## AddNumbers – S2 (Λεκτική Περιγραφή 3)

$$\begin{array}{r} 382 \\ + 835 \\ \hline \end{array}$$

- Πρόσθεσε το πρώτο δεξιά ψηφίο του 1<sup>ου</sup> αριθμού με το πρώτο δεξιά του 2<sup>ου</sup>
  - Αν το άθροισμα είναι μικρότερο από 9
    - Ενημέρωσε το τελικό άθροισμα με το ψηφίο του αθροίσματος
  - Αλλιώς
    - Ενημέρωσε το τελικό άθροισμα με το τελευταίο δεξιά ψηφίο του αθροίσματος
    - Κράτα ως κρατούμενο το αριστερό ψηφίο του αθροίσματος
- Για κάθε επόμενης τάξης ψηφίο**
- Πρόσθεσε το επόμενο ψηφίο του 1ου αριθμού με το επόμενο του 2ου
  - Αν υπάρχει από προηγούμενη πρόσθεση κρατούμενο
    - πρόσθεσε το
  - Ενημέρωσε το τελικό άθροισμα με τα ψηφία του αθροίσματος
  - Εμφάνισε το τελικό άθροισμα

3<sup>η</sup> εκδοχή

Συνεχίστε τον έλεγχο για διάφορα σενάρια.

## AddNumbers – S2 (Λεκτική Περιγραφή 4)

382  
835  
+123  
-----

- Πρόσθεσε το πρώτο δεξιά ψηφίο του 1<sup>ου</sup> αριθμού με το πρώτο δεξιά
- Αν το άθροισμα είναι μικρότερο από 9
  - Ενημέρωσε το τελικό άθροισμα με το ψηφίο του αθροίσματος
- Αλλιώς
  - Ενημέρωσε το τελικό άθροισμα με το τελευταίο δεξιά ψηφίο του αθροίσματος
  - Κράτα ως κρατούμενο το αριστερό ψηφίο του αθροίσματος
- **Για κάθε επόμενης τάξης ψηφίο**
  - Πρόσθεσε το επόμενο ψηφίο του 1ου αριθμού με το επόμενο του 2ου
  - Αν υπάρχει από προηγούμενη πρόσθεση κρατούμενο
    - πρόσθεσε το
  - Ενημέρωσε το τελικό άθροισμα με τα ψηφία του αθροίσματος
- Εμφάνισε το τελικό άθροισμα

4<sup>η</sup> εκδοχή

Συνεχίστε τον έλεγχο για διάφορα σενάρια.

## AddNumbers – S2 (Λεκτική Περιγραφή 5)

4382  
835  
746  
+ 23  
-----

- Θεώρησε τον μεγαλύτερο αριθμό πρώτο.
- για **κάθε ψηφίο του πρώτου αριθμού** ξεκινώντας από το λιγότερο σημαντικό
  - Υπολόγισε το άθροισμα των αντίστοιχης τάξης ψηφίων **όλων των αριθμών**
  - Αν υπάρχει από προηγούμενη πρόσθεση κρατούμενο
    - πρόσθεσε το στο άθροισμα των ψηφίων
  - Αν το άθροισμα των ψηφίων είναι μικρότερο από 9
    - Ενημέρωσε το τελικό άθροισμα με το άθροισμα των ψηφίων
  - Αλλιώς
    - Ενημέρωσε το τελικό άθροισμα με το λιγότερο σημαντικό ψηφίο του αθροίσματος των ψηφίων
    - Ενημέρωσε το κρατούμενο
- Εμφάνισε το τελικό άθροισμα.

5<sup>η</sup> εκδοχή

Συνεχίστε τον έλεγχο για διάφορα σενάρια.

## WordsHandling – Δείκτες και Δομές (struct)

```
160 void wordStats(void){
161     int curWord_len,max_len, min_len,total_len=0;
162     int i;
163
164     initializeWordStats(words[0]);
165     for(i=1;i<freeEntry;i++){
166         updateWordStats(words[i]);
167     }
168     displayWordStats();
169 }
```

Διαμόρφωσε τις συναρτήσεις που καλεί η wordStats ώστε να λειτουργούν με τις μεταβλητές στατιστικών λέξης ως τοπικές στην wordStats.  
Δες τι επιλογές έχεις;

Δοκίμασε με δείκτες ανάλογα με την getIntV2

## How I feel when my code works



### Απαραίτητη Προϋπόθεση

- A) Κατανόηση του προς ανάθεση έργου (είσοδοι, έξοδοι, ...)
- B) γνώση του αλγορίθμου,
- Γ) δυνατότητα περιγραφής του στην ομιλούμενη γλώσσα
- Δ) δυνατότητα μετατροπής της λεκτικής περιγραφής σε source code
- E) παραγωγή εκτελέσιμου κώδικα