

Εργαστήριο #4: Δομές επανάληψης

Απαιτήσεις:

- Απαγορεύονται **αυστηρά** η χρήση goto και η χρήση πινάκων.
- Χρησιμοποιήστε την καταλληλότερη δομή επανάληψης για κάθε περίπτωση.

Άσκηση 1

Αποθηκεύστε το πρόγραμμά σας σε αρχείο με όνομα **lab4a.c**

Ζητούμενα: Γράψτε ένα πρόγραμμα το οποίο αρχικά διαβάζει ένα χρονικό διάστημα κατά το οποίο μία αίθουσα είναι δεσμευμένη για μαθήματα (σε όλες τις άλλες ώρες του 24ώρου είναι ελεύθερη). Στη συνέχεια διαβάζει ένα νέο αίτημα δέσμευσης κι ενημερώνει τον χρήστη κατά πόσο η αίθουσα είναι διαθέσιμη στις ζητούμενες ώρες. Υποθέστε ότι οι τιμές εισόδου είναι εγγυημένα στο διάστημα [0, 24].

Παράδειγμα: Αν η αίθουσα είναι αρχικά δεσμευμένη από τις 5 έως τις 8, τότε για ένα αίτημα δέσμευσης από τις 3 έως τις 5, δεν υπάρχει επικάλυψη. Αίτημα από τις 4 μέχρι τις 7 ή από τις 2 μέχρι τις 9, προκαλεί μερική επικάλυψη, ενώ από τις 7 μέχρι τις 8 πλήρη επικάλυψη.

Το πρόγραμμα λειτουργεί ως εξής:

1. Εκτυπώνει το μήνυμα **"Current reservation:\n"** και διαβάζει το χρονικό διάστημα της ήδη υπάρχουσας δέσμευσης της αίθουσας στη μορφή **S-E** όπου **S** η ώρα έναρξης και **E** η ώρα λήξης (η αίθουσα είναι ελεύθερη από την ώρα **E** και μετά). Όσο η ώρα έναρξης δεν προηγείται της ώρας λήξης, το βήμα 1 επαναλαμβάνεται.
2. Σε επανάληψη,
 - Εκτυπώνει το μήνυμα **"#\nRequest:\n"** και διαβάζει ένα διάστημα στη μορφή **X-Y** όπου **X** και **Y** η επιθυμητή ώρα έναρξης και λήξης αντίστοιχα. Όπου βλέπετε **X** και **Y** παρακάτω, θεωρήστε την ίδια αντιστοιχία¹. Στα μηνύματα εκτυπώνετε τις αντίστοιχες ώρες και όχι τα γράμματα **X** και **Y**.
 - Αν και οι δύο ώρες έχουν τιμή μηδέν, η επανάληψη τερματίζει.
 - Αν η ώρα έναρξης δεν προηγείται της ώρας λήξης, εκτυπώνει το μήνυμα **"X-Y invalid.\n"**, και προχωρά στην επόμενη επανάληψη.
 - Διαφορετικά, γίνονται οι παρακάτω έλεγχοι (σκεφτείτε με τι σειρά είναι πιο πρακτικό να γίνουν):
 - Αν το διάστημα **[X, Y)** περιέχεται πλήρως μέσα στο **[S, E)** εκτυπώνει το μήνυμα **"X-Y full overlap.\n"**
 - Αν υπάρχει μερική επικάλυψη ανάμεσα στα **[X, Y)** και **[S, E)**, εκτυπώνει το μήνυμα **"X-Y partial overlap.\n"**
 - Αν δεν υπάρχει επικάλυψη, εκτυπώνει το μήνυμα **"X-Y no overlap.\n"** και η επανάληψη τερματίζει.
3. Εκτυπώνει το μήνυμα **"#\nInvalid I. Valid V. Partial P. Full F.\n"** όπου **I** το πλήθος άκυρων (invalid) διαστημάτων από το βήμα 2, **V** το πλήθος έγκυρων διαστημάτων (μη-invalid και μη-μηδενικών), **P** το πλήθος διαστημάτων με μερική επικάλυψη και **F** το πλήθος διαστημάτων με πλήρη επικάλυψη.

¹ ΜΗ χρησιμοποιήσετε ονόματα μεταβλητών **x, y, s, e** γιατί δεν είναι περιγραφικά!

Άσκηση 2

Αποθηκεύστε το πρόγραμμά σας σε αρχείο με όνομα **lab4b.c**

Ζητούμενα: Γράψτε ένα πρόγραμμα το οποίο εκτυπώνει ένα ορθογώνιο με εναλλάξ χαρακτήρες.

Το πρόγραμμα λειτουργεί ως εξής:

1. Εκτυπώνει `"Enter rows and columns [1-20]:\n"` και διαβάζει δύο ακέραιους: πλήθος γραμμών και πλήθος στηλών. Όσο αυτοί δεν ανήκουν στο διάστημα [1, 20], το βήμα 1 επαναλαμβάνεται.
2. Εκτυπώνει `"Enter 2 distinct characters:\n"` και διαβάζει 2 χαρακτήρες χωρίς κενό ανάμεσα τους. Όσο αυτοί είναι ίδιοι, εκτυπώνει `"Try again!\n"` και το βήμα 2 επαναλαμβάνεται.
3. Με χρήση αποκλειστικά της συνάρτησης `putchar` (όχι `printf` σε αυτό το βήμα!) εμφανίζει στην οθόνη ένα σχήμα που σχηματίζεται από τους δύο χαρακτήρες ως εξής: Κάθε γραμμή έχει τόσες εμφανίσεις του κάθε χαρακτήρα όσο το πλήθος στηλών, τοποθετώντας τους εναλλάξ, χωρίς κενό ανάμεσα, και **χαρακτήρα αλλαγής γραμμής** στο τέλος. Η πρώτη γραμμή ξεκινά από τον πρώτο χαρακτήρα που δόθηκε στο βήμα 2 και κάθε επόμενη γραμμή ξεκινά από τον χαρακτήρα που είναι διαφορετικός από αυτόν που ξεκίνησε η προηγούμενη.

Παράδειγμα: Για εισόδους 4 5 στο βήμα 1 και -X στο βήμα 2, θα εμφανιστεί στην οθόνη:

```
-X-X-X-X-X
X-X-X-X-X-
-X-X-X-X-X
X-X-X-X-X-
```

Σκεφτείτε έναν έξυπνο τρόπο να υλοποιήσετε την εναλλαγή χαρακτήρων.

Υποβολή

Κατασκευάστε εκ νέου την ομάδα σας στο **autolab**.

Ανοίξτε ένα τερματικό και μεταβείτε στον κατάλογο που βρίσκονται τα αρχεία `lab4a.c` και `lab4b.c`. Εκτελέστε την εντολή:

```
zip -r lab4.zip lab4a.c lab4b.c
```

Θα πρέπει να δημιουργηθεί ένα αρχείο με όνομα `lab4.zip`

Επιβεβαιώστε ότι τα περιεχόμενά του είναι σωστά γράφοντας την εντολή:

```
zip -sf lab4.zip
```

Θα πρέπει να σας εμφανίσει κάτι σαν το παρακάτω:

```
Archive contains:
  lab4a.c
  lab4b.c
Total 2 entries
```

Υποβάλετε στο **autolab** το αρχείο `lab4.zip`.