

Προγραμματιστική Άσκηση 1

Ημερομηνία Παράδοσης: Τετάρτη, 18 Μαρτίου 2009

Πρόβλημα 1 [10 μονάδες] Το ζητούμενο αυτής της προγραμματιστικής άσκησης είναι να γράψετε ένα πρόγραμμα σε C το οποίο δεδομένου ενός συνόλου μη προσημασμένων ακεραίων αριθμών να το διαμερίζει σε υποσύνολα διαδοχικών μη προσημασμένων ακεραίων. Στη συνέχεια όταν αναφερόμαστε σε ακεραίους θα εννοούμε μη προσημασμένους ακεραίους. Πιο συγκεκριμένα, αν S κάποιο σύνολο ακεραίων αριθμών, θέλουμε να κατασκευάσουμε τη διαμέριση $\{S_1, S_2, \dots, S_M\}$ του S έτσι ώστε:

- $\forall i \in S, i \in S_k$, για κάποιο $k, 1 \leq k \leq M$, και $i \notin S_m$, για $k \neq m$ (δηλαδή κάθε στοιχείο του S πρέπει να ανήκει σε ακριβώς ένα υποσύνολο του S εκ των S_1, S_2, \dots, S_M , το οποίο επίσης σημαίνει ότι η ένωση των υποσυνόλων αυτό είναι ίση με το S).
- Αν $i, i+1 \in S$, τότε $i, i+1 \in S_k$, για κάποιο k (δηλαδή, αν το S περιέχει δύο διαδοχικούς ακεραίους τότε και οι δύο πρέπει να ανήκουν στο ίδιο υποσύνολο εκ των S_1, S_2, \dots, S_M).
- Αν $i, j \in S$ και $|i - j| > 1$, τότε $i \in S_k$ και $j \in S_m$, όπου $k \neq m$.

Για παράδειγμα, αν μας δίνεται το σύνολο ακεραίων $S = \{0, 2, 3, 4, 7, 10, 23, 24, 34, 36, 37\}$, η διαμέριση που ζητάμε αποτελείται από 7 υποσύνολα του S , και ειδικότερα:

$$\begin{aligned} S_1 &= \{0\}, \\ S_2 &= \{2, 3, 4\}, \\ S_3 &= \{7\}, \\ S_4 &= \{10\}, \\ S_5 &= \{23, 24\}, \\ S_6 &= \{34\}, \\ S_7 &= \{36, 37\}. \end{aligned}$$

Στην προγραμματιστική άσκηση αυτή ένα σύνολο ακεραίων αριθμών n στοιχείων θα αναπαρίσταται μέσω ενός πίνακα ακεραίων αριθμών έτσι ώστε το στοιχείο στη θέση 0 του πίνακα να είναι ίσο με το πλήθος n των στοιχείων του συνόλου, ενώ στις θέσεις 1 ως n θα έχουμε τα στοιχεία του συνόλου, ταξινομημένα σε αύξουσα σειρά. Για παράδειγμα το σύνολο $\{2, 3, 5, 8, 9\}$ αναπαρίστανται από ένα πίνακα a έξι ακεραίων αριθμών, και $a[0] = 5, a[1] = 2, a[2] = 3, a[3] = 5, a[4] = 8$ και $a[5] = 9$, ή σχηματικά:

a :

5	2	3	5	8	9
---	---	---	---	---	---

Για την 1η Προγραμματιστική Άσκηση σας δίνεται το αρχείο `split.c` το οποίο περιέχει ήδη κάποια κομμάτια κώδικα σε C, και το οποίο πρέπει να συμπληρώσετε με το δικό σας κώδικα. Ειδικότερα θα πρέπει να συμπληρώσετε τη συνάρτηση `split` με prototype

```
unsigned int **split(unsigned int *s, unsigned int *nsets);
```

η οποία δεδομένου ενός συνόλου ακεραίων s επιστρέφει πίνακα σε σύνολα ακεραίων, ενώ το πλήθος των συνόλων που επιστρέφει αποθηκεύεται στη μεταβλητή $*nsets$. Τα σύνολα ακεραίων που επιστρέφονται θα πρέπει να δεσμευθούν στη μνήμη δυναμικά με χρήση, π.χ., της συνάρτησης `malloc`.

Προκειμένου να έχετε μια γενικότερη εποπτεία του τί κάνει το πρόγραμμά σας θα εξηγηθεί παρακάτω η συνολική συμπεριφορά του προγράμματός σας. Πέραν του υπολογισμού της διαμέρισης όλα τα υπόλοιπα στοιχεία του προγράμματος έχουν υλοποιηθεί για λογαριασμό σας.

Το πρόγραμμά σας θα πρέπει να διαβάζει αρχείο με σύνολα ακεραίων αριθμών και στη συνέχεια να υπολογίζει τη διαμέριση καθενός από τα σύνολα που διάβασε. Κάθε αρχείο δεδομένων μπορεί να περιέχει ένα ή παραπάνω σύνολα ακεραίων αριθμών. Στην πρώτη γραμμή το αρχείο περιέχει το πλήθος των συνόλων που περιέχει συνολικά το αρχείο. Σε κάθε επόμενη γραμμή καταγράφεται ένα σύνολο ακεραίων αριθμών ανά γραμμή, με πρώτο να καταγράφεται το μέγεθος του συνόλου και στη συνέχεια ένα-ένα τα στοιχεία του σε αύξουσα σειρά. Για παράδειγμα, στον παρακάτω πίνακα παρατίθενται τα περιεχόμενα του αρχείου `ex1.cin` (ενός εκ των αρχείων που σας δίνονται), το οποίο περιέχει δύο σύνολα ακεραίων αριθμών (γι' αυτό και υπάρχει το 2 στην πρώτη γραμμή του αρχείου), και πιο συγκεκριμένα τα σύνολο $\{0, 2, 3, 4, 7, 10, 23, 24, 34, 36, 37\}$ που αποτελείται από 11 στοιχεία (δεύτερη γραμμή του αρχείου) και το σύνολο $\{3, 4, 5, 6, 7, 28, 29, 30, 31, 60, 61, 105, 106, 107, 108, 109, 169\}$ που αποτελείται από 17 στοιχεία (τρίτη γραμμή του αρχείου).

```
2
11 0 2 3 4 7 10 23 24 34 36 37
17 3 4 5 6 7 28 29 30 31 60 61 105 106 107 108 109 169
```

Το αρχείο από το οποίο θα διαβάσει το πρόγραμμά σας δίνεται ως `command-line argument` (ήδη έχει υλοποιηθεί για εσάς). Μία τυπική εκτέλεση του προγράμματός σας για τα δεδομένα του αρχείου `ex1.cin` δίνεται στον παρακάτω πίνακα (τα δεδομένα που δίνει ο χρήστης γράφονται με κόκκινο, το `prompt` με μαύρο ενώ ότι εκτυπώνει το πρόγραμμα με μπλε).

```
[my prompt]$ split ex1.cin ←
Set #1: {0, 2, 3, 4, 7, 10, 23, 24, 34, 36, 37}:
{{0}, {2, 3, 4}, {7}, {10}, {23, 24}, {34}, {36, 37}}

Set #2: {3, 4, 5, 6, 7, 28, 29, 30, 31, 60, 61, 105, 106, 107, 108, 109, 169}:
{{3, 4, 5, 6, 7}, {28, 29, 30, 31}, {60, 61}, {105, 106, 107, 108, 109}, {169}}

[my prompt]$
```

Πέραν του αρχείου `split.c` το οποίο και πρέπει να συμπληρώσετε σας δίνονται επίσης τρία αρχεία δεδομένων `ex1.cin`, `ex2.cin`, `ex3.cin`, προκειμένου να ελέγξετε τα αποτελέσματά σας. Τα αρχεία αυτά περιέχουν αντίστοιχα 2, 10 και 100 σύνολα ακεραίων αριθμών.

Τα αρχεία που σας δίδονται μπορείτε να τα βρείτε στην ιστοσελίδα του μαθήματος στο URL:

<http://www.tem.uoc.gr/~mkaravel/Courses/EM102-spring09/hwks/hwk1-files.zip>

Από εσάς ζητάται να αλλάξετε κατάλληλα το αρχείο `split.c` και να το επιστρέψετε. Θα ανακοινωθεί στην ιστοσελίδα του μαθήματος πώς θα επιστρέψετε το πρόγραμμά σας.

Στο πρόγραμμά σας θα πρέπει να γράψετε σε σχόλια το ονοματεπώνυμό σας και τον αριθμό μητρώου σας, καθώς και ότι άλλες πληροφορίες είναι σκόπιμες προκειμένου όποιος διαβάσει το πρόγραμμά σας να καταλάβει τί γίνεται.

Ο βαθμός σας για την άσκηση αυτή χωρίζεται στις εξής κατηγορίες:

- Την αναγνωσιμότητα του κώδικά σας.
- Κάλυψη όλων των δυνατών περιπτώσεων.
- Την ορθότητα του κώδικά σας (θα πρέπει να δίνει τα σωστά αποτελέσματα).
- Τήρηση των προδιαγραφών που αναφέρονται στην εκφώνηση
- Από την απλότητα του κώδικά σας και του τρόπου υλοποίησης της `split`.

Σύνολο μονάδων: 10