

ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ (2020-21)

Εργασία 1

Ένας ακέραιος αριθμός μεγαλύτερος του 1 ονομάζεται *πρώτος* (prime) όταν έχει σαν μόνους διαιρέτες το 1 και τον εαυτό του. Για παράδειγμα, το 13 είναι πρώτος αριθμός, ενώ το 15 ($= 3 \times 5$) δεν είναι.

Παραγοντοποίηση (factorization) ενός ακεραίου αριθμού μεγαλύτερου του 1 είναι η εύρεση των πρώτων αριθμών που το γινόμενό τους ισούται με τον αριθμό. Μερικά παραδείγματα παραγοντοποίησης είναι τα εξής:

$$\begin{aligned} 6 &= 2 \times 3 \\ 45 &= 3 \times 3 \times 5 \\ 187 &= 11 \times 17 \\ 5234 &= 2 \times 2617 \\ 10201 &= 101 \times 101 \\ 128377 &= 128377 \\ 5489126 &= 2 \times 2744563 \\ 89000362 &= 2 \times 11 \times 29 \times 199 \times 701 \\ 127870872 &= 2 \times 2 \times 2 \times 3 \times 17 \times 313409 \\ 2100980393 &= 45613 \times 46061 \end{aligned}$$

Ένας αριθμός ονομάζεται *ημιπρώτος* (semiprime) όταν έχει ακριβώς δύο πρώτους παράγοντες, όχι κατ' ανάγκη διαφορετικούς μεταξύ τους.¹ Από τα παραπάνω παραδείγματα αριθμών, οι 6, 187, 5234, 10201, 5489126 και 2100980393 είναι ημιπρώτοι.

Γενίκευση της έννοιας των ημιπρώτων αριθμών είναι οι *k-σχεδόν πρώτοι* (*k-almost primes*), οι οποίοι έχουν ακριβώς *k* πρώτους παράγοντες. Προφανώς, η έννοια των ημιπρώτων ταυτίζεται με αυτή των 2-σχεδόν πρώτων και οι πρώτοι αριθμοί είναι οι 1-σχεδόν πρώτοι. Από τα παραπάνω παραδείγματα αριθμών, ο 128377 είναι πρώτος (1-σχεδόν πρώτος), ο 45 είναι 3-σχεδόν πρώτος, ο 89000362 είναι 5-σχεδόν πρώτος και ο 127870872 είναι 6-σχεδόν πρώτος.

Τέλος, ορίζουμε και τους *ενδιαφέροντες ημιπρώτους* (interesting semiprimes) οι οποίοι είναι οι ημιπρώτοι των οποίων οι δύο πρώτοι παράγοντες είναι άνισοι, αλλά ο μικρότερος από αυτούς δεν απέχει από τον μεγαλύτερο περισσότερο από το 1% του μεγαλύτερου. Από τα παραπάνω παραδείγματα αριθμών, μόνο ο 2100980393 είναι ενδιαφέρων ημιπρώτος.

Γράψτε, κατ' αρχήν, ένα πρόγραμμα C (έστω ότι το πηγαίο αρχείο του ονομάζεται "kalmpr.c") το οποίο να υπολογίζει και να εκτυπώνει το πλήθος όλων των *k-σχεδόν πρώτων* αριθμών που βρίσκονται μέσα στο διάστημα $[\text{MINNUM}, \text{MAXNUM}]$, για $2 \leq k \leq \text{MAXK}$, καθώς και το πλήθος αυτών που είναι, ειδικότερα, 2-σχεδόν πρώτοι, δηλαδή ημιπρώτοι. Για τους ενδιαφέροντες ημιπρώτους, να εκτυπώνονται και οι παράγοντες στους οποίους αναλύονται. Οι συμβολικές σταθερές MINNUM, MAXNUM και MAXK θα πρέπει να οριστούν στο πρόγραμμά σας με #define.

Μία ενδεικτική εκτέλεση του προγράμματος, για MINNUM ίσο με 100, MAXNUM ίσο με 100000 και MAXK ίσο με 4 φαίνεται στη συνέχεια:

```
$ ./kalmpr
Checking numbers in [100,100000]

227 * 229 = 51983 is an interesting semiprime
```

¹Σημειώνεται ότι οι ημιπρώτοι αριθμοί έχουν ιδιαίτερη σημασία και χρησιμότητα στην περιοχή της κρυπτογραφίας.

```
239 * 241 = 57599 is an interesting semiprime
269 * 271 = 72899 is an interesting semiprime
281 * 283 = 79523 is an interesting semiprime
311 * 313 = 97343 is an interesting semiprime
```

Found 67611 k -almost prime numbers ($2 \leq k \leq 4$)

Found 23344 semiprimes

\$

Στη συνέχεια, αυξάνετε διαδοχικά το εύρος του διαστήματος `[MINNUM,MAXNUM]`, για να διαπιστώσετε τον ρυθμό με τον οποίο αυξάνει και ο χρόνος εκτέλεσης του προγράμματός σας.

Τέλος, επεκτείνετε το πρόγραμμά σας έτσι ώστε να επιλέγει με τυχαίο τρόπο 20 θετικούς ακεραίους αριθμούς (όχι κατ' ανάγκη διαφορετικούς μεταξύ τους) για καθένα από του οποίους να εκτυπώνει το k που του αντιστοιχεί. Είναι προφανές ότι κάθε αριθμός είναι k -σχεδόν πρώτος, για συγκεκριμένο k . Το πλήθος των αριθμών που θα ελεγχθούν (20) να μην τοποθετηθεί απ' ευθείας μέσα στο πρόγραμμά σας, αλλά να ορισθεί μέσω `#define` σαν τιμή της συμβολικής σταθεράς `COMPUTATIONS`. Ο τυχαίος τρόπος επιλογής των αριθμών που θα ελεγχθούν πρέπει να ακολουθεί τη διαδικασία που περιγράφεται στη συνέχεια.

Στην C μπορούμε να “γεννήσουμε” τυχαίους αριθμούς με χρήση των συναρτήσεων `srand` και `rand`, όπως επιδεικνύεται στο πρόγραμμα <http://www.di.uoa.gr/~ip/cprogs/gcdlcm.c>, στις σελίδες 44–45 των διαφανειών του μαθήματος, στο <http://www.di.uoa.gr/~ip/K04.pdf>.

Έστω ότι γεννάτε δύο τυχαίους αριθμούς y και z με χρήση δύο διαδοχικών κλήσεων της συνάρτησης `rand`. Ο αριθμός x για τον οποίο θα πρέπει να βρείτε το k που του αντιστοιχεί δίνεται από τον τύπο²

$$x = ((y \bmod 32768) + 1) \cdot ((z \bmod 32768) + 1) + 1$$

Για την αρχικοποίηση της γεννήτριας των τυχαίων αριθμών, να χρησιμοποιήσετε την τρέχουσα ώρα, μέσω της συνάρτησης `time`, όπως ακριβώς γίνεται και στο πρόγραμμα `gcdlcm.c` που προαναφέρθηκε.

Μία ενδεικτική εκτέλεση³ του προγράμματος φαίνεται στη συνέχεια:

\$./kalmpr

Checking numbers in [2100000000,2101000000]

```
45767 * 45887 = 2100110329 is an interesting semiprime
45821 * 45833 = 2100113893 is an interesting semiprime
45667 * 45989 = 2100179663 is an interesting semiprime
45677 * 45979 = 2100182783 is an interesting semiprime
45697 * 45959 = 2100188423 is an interesting semiprime
45707 * 45949 = 2100190943 is an interesting semiprime
45763 * 45893 = 2100201359 is an interesting semiprime
45823 * 45833 = 2100205559 is an interesting semiprime
45631 * 46027 = 2100258037 is an interesting semiprime
45817 * 45841 = 2100297097 is an interesting semiprime
45707 * 45953 = 2100373771 is an interesting semiprime
45767 * 45893 = 2100384931 is an interesting semiprime
45827 * 45833 = 2100388891 is an interesting semiprime
45613 * 46049 = 2100433037 is an interesting semiprime
```

²Με `mod` συμβολίζουμε το υπόλοιπο διαίρεσης.

³Η συγκεκριμένη εκτέλεση έγινε σε μηχανήμα Linux του εργαστηρίου του Τμήματος. Σε άλλες πλατφόρμες (π.χ. Windows, MacOS) τα αποτελέσματα ενδέχεται να διαφέρουν, διότι η γεννήτρια τυχαίων αριθμών μπορεί να παράγει διαφορετική ακολουθία αριθμών, ακόμα και με το ίδιο “φύτρο”.

```
45641 * 46021 = 2100444461 is an interesting semiprime
45673 * 45989 = 2100455597 is an interesting semiprime
45691 * 45971 = 2100460961 is an interesting semiprime
45821 * 45841 = 2100480461 is an interesting semiprime
45613 * 46051 = 2100524263 is an interesting semiprime
45823 * 45841 = 2100572143 is an interesting semiprime
45677 * 45989 = 2100639553 is an interesting semiprime
45707 * 45959 = 2100648013 is an interesting semiprime
45779 * 45887 = 2100660973 is an interesting semiprime
45641 * 46027 = 2100718307 is an interesting semiprime
45697 * 45971 = 2100736787 is an interesting semiprime
45827 * 45841 = 2100755507 is an interesting semiprime
45691 * 45979 = 2100826489 is an interesting semiprime
45817 * 45853 = 2100846901 is an interesting semiprime
45779 * 45893 = 2100935647 is an interesting semiprime
45613 * 46061 = 2100980393 is an interesting semiprime
```

Found 745581 k-almost prime numbers ($2 \leq k \leq 5$)

Found 151473 semiprimes

Current time is 1604430088

Checking 20 random numbers

```
1860481 is a 4-almost prime number
70979693 is a 2-almost prime number
789586741 is a 2-almost prime number
92273518 is a 4-almost prime number
120993961 is a 4-almost prime number
187268828 is a 4-almost prime number
37267461 is a 5-almost prime number
551604007 is a 2-almost prime number
55414001 is a 3-almost prime number
397964493 is a 3-almost prime number
16620311 is a 2-almost prime number
427050025 is a 3-almost prime number
989544503 is a 4-almost prime number
131816784 is a 8-almost prime number
77583831 is a 3-almost prime number
118887301 is a 3-almost prime number
185517113 is a 3-almost prime number
144277971 is a 3-almost prime number
229923941 is a 3-almost prime number
190987964 is a 4-almost prime number
$
```

Σημειώσεις/Απαγορεύσεις:

1. Στο πρόγραμμα που θα παραδώσετε, να θέσετε τέτοιες τιμές στις συμβολικές σταθερές, ώστε αυτό να τερματίζει σε λιγότερο από ένα λεπτό σε μηχάνημα Linux του εργαστηρίου του Τμήματος.

2. Για να χρονομετρήσετε το πρόγραμμά σας, μπορείτε να το καλέσετε (σε μηχανήμα Linux) ως `time ./kalmpr`. Ο πρώτος αριθμός που θα εκτυπωθεί από την `time` είναι ο χρόνος, σε δευτερόλεπτα, που αφιέρωσε ο επεξεργαστής του υπολογιστή σας στο πρόγραμμά σας. Εναλλακτικά, μπορείτε να χρησιμοποιήσετε τη συνάρτηση `clock`, τη χρήση της οποία μπορείτε να δείτε στο πρόγραμμα <http://www.di.uoa.gr/~ip/cprogs/sorting.c>, στις σελίδες 167–169 των διαφανειών του μαθήματος.
3. Αν θέλετε να συγκρίνετε τα αποτελέσματά σας με αυτά της ενδεικτικής εκτέλεσης που δόθηκε προηγουμένως, μπορείτε να αρχικοποιήσετε την γεννήτρια τυχαίων αριθμών όχι με τον τρέχοντα χρόνο, αλλά με τη συγκεκριμένη τιμή που φαίνεται στην ενδεικτική εκτέλεση και, φυσικά, να δουλέψετε σε μηχανήμα Linux του εργαστηρίου του Τμήματος.
4. Στην εργασία αυτή **απαγορεύονται αυστηρά η χρήση πινάκων, δεικτών και συναρτήσεων της μαθηματικής βιβλιοθήκης της C**. Επίσης, **δεν επιτρέπεται ο ορισμός άλλων συναρτήσεων πλην της `main()`**.
5. Η παράδοση της εργασίας αυτής συνίσταται στην υποβολή του πηγαίου αρχείου `kalmpr.c` μέσω του e-class του μαθήματος (επιλογή “Εργασίες”).