ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ - ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ
Ειδικά Θέματα Αλγορίθμων
Διδάσκων: Ε. Μαρκάκης, Φθινοπωρινό εξάμηνο 2023

1η σειρά ασκήσεων **Προθεσμία παράδοσης: 11 Δεκεμβρίου 2023**

Πρόβλημα 1. (13 μονάδες)

(i) (6 μονάδες) Θεωρήστε την ακολουθία των αριθμών Fibonacci, με $F_n = F_{n-1} + F_{n-2}$, $F_1 = 1$, $F_0 = 0$. Απόδειξτε ότι ισχύει η παρακάτω σχέση για κάθε $k \geq 1$.

$$F_{n+k} = F_{n+1} \cdot F_k + F_n \cdot F_{k-1}$$

(ii) (7 μονάδες) Εφαρμόστε το Κινέζικο θεώρημα υπολοίπων για να βρείτε όλες τις λύσεις του συστήματος $x \equiv 3 \ mod \ 4, x \equiv 1 \ mod \ 5, x \equiv 2 \ mod \ 7$. Να χρησιμοποιήσετε τον εκτεταμένο αλγόριθμο του Ευκλείδη για τους υπολογισμούς των πολλαπλασιαστικών αντιστρόφων που θα χρειαστείτε.

Πρόβλημα 2. (18 μονάδες)

(i) (4 μονάδες) Υπολογίστε το $\phi(243)$.

(ii) (6 μονάδες) Χωρίς να χρησιμοποιήσετε κομπιουτεράκι ή οποιοδήποτε άλλο μέσο (ούτε τον αλγόριθμο επαναλαμβανόμενου τετραγωνισμού), υπολογίστε τις ποσότητες: $7 \cdot 3^{58} \ mod \ 19$, και $(3 \cdot 7^{17} + 11^{33} + 3 \cdot 13^{49}) \ mod \ 60$.

(iii) (8 μονάδες) Θεωρήστε το Fermat test που παρουσιάστηκε στις διαφάνειες σαν ένας αλγόριθμος για το primality testing. Έστω ότι ένας αριθμός n είναι σύνθετος και δεν είναι αριθμός Carmichael. Έστω επίσης ότι για να τρέξετε το Fermat test, επιλέγετε τυχαία έναν αριθμό α, έτσι ώστε $\alpha \in \{1, \ldots, n-1\}$ και $gcd(\alpha, n) = 1$. Να δείξετε ότι τουλάχιστον για τις μισές από τις πιθανές επιλογές για το α, ο αριθμός n δεν θα περάσει το Fermat test (δηλαδή θα αναγνωρισθεί ως σύνθετος).

Πρόβλημα 3. (10 μονάδες) Έστω ότι σε ένα σύστημα RSA, ο Oscar υποκλέπτει το ciphertext, το οποίο είναι ίσο με $C = 10$. Αν ξέρει ότι το ciphertext αυτό είχε σταλεί σε ένα χρήστη με public key $e = 13$, και $n = 35$, ποιο ήταν το plaintext? Δείξτε αναλυτικά όλα τα βήματα που πρέπει να ακολουθήσετε για να βρείτε το μήνυμα. Αν χρειαστεί να υψώσετε σε δύναμη για τους υπολογισμούς που θα κάνετε, θα πρέπει να χρησιμοποιήσετε τον αλγόριθμο επαναλαμβανόμενου τετραγωνισμού.

Πρόβλημα 4. (14 μονάδες) Έστω ένας πίνακας Α n θέσεων, από το 1 ως το n. Θεωρήστε τον παρακάτω ψευδοκώδικα, που παίρνει ως είσοδο έναν ακέραιο m με $1 \leq m \leq n$.

```
int foo(int m)
if (m ≤ √n) return n + 10
else {
    int x=0
```

for $i = 1$ to n $\{x = x + A[i]\}$
return x
}

(i) (7 μονάδες) Αναλύστε την πολυπλοκότητα καλύτερης, χειρότερης και μέσης περίπτωσης. Για την ανάλυση της μέσης περίπτωσης, θεωρήστε ότι ο ακέραιος m επιλέγεται τυχαία με ομοιόμορφη κατανομή από τους ακεραίους $1, 2, \ldots, n$.

(ii) (7 μονάδες) Απαντήστε στα ίδια ερωτήματα αν τώρα αντικαταστήσουμε την συνθήκη του if με if ($m \leq n/2$).

Πρόβλημα 5. (10 μονάδες) Θεωρήστε τους παρακάτω περιορισμούς ενός γραμμικού προγράμματος.

$$\begin{aligned} -x_1 \ + \ 3x_2 \ &\leq 30 \\ -3x_1 \ + \ x_2 \ &\leq 30 \\ x_1, x_2 &\geq 0 \end{aligned}$$

(i) (4 μονάδες) Να σχεδιάσετε την εφικτή περιοχή και να αποφασίσετε αν είναι φραγμένη ή μη φραγμένη.

(ii) (6 μονάδες) Έστω ότι η αντικειμενική συνάρτηση που θέλουμε να μεγιστοποιήσουμε είναι η $f(x_1, x_2) = -\frac{1}{2}x_1 - x_2$. Χρησιμοποιήστε την γραφική μέθοδο για να αποφασίσετε αν το γραμμικό πρόγραμμα έχει βέλτιστη λύση. (**Tip**: Για την καλύτερη επίδειξη της μεθόδου, ξεκινήστε με την τιμή $Z = -30$ ή κάποια άλλη κοντινή τιμή και χρησιμοποιήστε τουλάχιστον άλλες 2 τιμές).

Πρόβλημα 6. (10 μονάδες) Διατυπώστε τα παρακάτω δύο προβλήματα σαν ακέραια προγράμματα.

(i) (4 μονάδες) Θέλετε να αγοράσετε σοκολατένια μπισκότα και έχετε στην διάθεσή σας 35 ευρώ. Στο περίπτερο της γειτονιάς σας, βλέπετε ότι υπάρχουν μόνο 5 συσκευασίες με μπισκότα προς πώληση. Οι πρώτες δύο κοστίζουν 10 ευρώ η κάθεμια, ενώ οι υπόλοιπες κοστίζουν 6, 14 και 11 ευρώ αντίστοιχα. Οι πρώτες 2 συσκευασίες έχουν 8 μπισκότα η κάθεμια, και οι υπόλοιπες έχουν 9, 7, και 10 μπισκότα αντίστοιχα. Ο στόχος σας είναι να μεγιστοποιήσετε τον αριθμό των μπισκότων που θα αγοράσετε χωρίς να ξοδέψετε παραπάνω από τα 35 ευρώ που διαθέτετε.

(ii) (6 μονάδες) Θεωρήστε ένα γράφο $G = (V, E)$ και έστω ότι κάθε κορυφή $v \in V$ έχει ένα βάρος $w_v \geq 0$. Ένα *ανεξάρτητο σύνολο* σε ένα γράφο είναι ένα υποσύνολο $S \subseteq V$, έτσι ώστε για κάθε $u, v \in S$, ισχύει ότι $(u, v) \notin E$, δηλαδή, ένα ανεξάρτητο σύνολο περιέχει κορυφές που δεν συνδέονται μεταξύ τους. Στόχος είναι η εύρεση του ανεξάρτητου συνόλου με το μεγαλύτερο συνολικό βάρος (με το μεγαλύτερο άθροισμα βαρών των κορυφών του).

Πρόβλημα 7. (10 μονάδες) Extra credit: Χρησιμοποιήστε την σχέση που αποδείξατε στο Πρόβλημα 1- (i) για να σχεδιάσετε έναν $O(logn)$ αλγόριθμο για τον υπολογισμό του n-οστού αριθμού Fibonacci. Επιχειρηματολογήστε αν ο αλγόριθμος αυτός είναι πιο γρήγορος στην πράξη από τον άλλο αλγόριθμο που έχουμε δει με πολυπλοκότητα $O(logn)$.