Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Προηγμένα Συστήματα Πληροφορικής»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**Μελέτη του Προβλήματος των Κίβδηλων Νομισμάτων**
Ονοματεπώνυμο Φοιτητή	**Προδρομίδου Μαρία-Άννα**
Πατρώνυμο	**Γεώργιος**
Αριθμός Μητρώου	**ΜΠΣΠ/ 09041**
Επιβλέπων	**Καθηγητής Παναγιώτης Τσικούρας**

Ημερομηνία Παράδοσης **Ιούνιος 2013**

Τριμελής Εξεταστική Επιτροπή

Παναγιώτης Τσικούρας Καθηγητής	Αριστείδης Σαπουνάκης Καθηγητής	Ευάγγελος Φούντας Καθηγητής

Πρόλογος

Η παρούσα μεταπτυχιακή διατριβή εκπονήθηκε στα πλαίσια του Μεταπτυχιακού Προγράμματος Σπουδών του Τμήματος Πληροφορικής “Προηγμένα Συστήματα Πληροφορικής”, του Πανεπιστημίου Πειραιώς. Στόχος αυτής της διπλωματικής είναι η μελέτη του προβλήματος των κίβδηλων νομισμάτων καθώς και η ανάπτυξη αλγορίθμων για την επίλυση του προβλήματος των κίβδηλων νομισμάτων με τον ελάχιστο αριθμό ζυγισμάτων.

Σε αυτό το σημείο, επιθυμώ να ευχαριστήσω θερμά τον επιβλέποντα της εργασίας μου, Καθηγητή Παναγιώτη Τσικούρα, που μου έδωσε την ευκαιρία να εκπονήσω την παρούσα διατριβή, καθώς και για την καθοδήγηση και την βοήθεια που μου προσέφερε καθ’ όλη τη διάρκεια της προσπάθειας αυτής. Επίσης, ευχαριστώ τα μέλη της τριμελούς επιτροπής Καθηγητές Αριστείδη Σαπουνάκη και Ευάγγελο Φούντα. Τέλος, θέλω να ευχαριστήσω τον διδάκτορα Γιάννη Τασούλα και τον υποψήφιο διδάκτορα Κώστα Μανέ για όλα όσα με διδάξανε, για το επιστημονικό υλικό που μου προσφέρανε, τη συμπαράστασή τους και τις ώρες που μου αφιερώσανε.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένειά μου για την στήριξη που μου προσέφερε καθ’ όλη τη διάρκεια των σπουδών μου.

Περίληψη

Η διπλωματική αυτή ασχολείται με το πρόβλημα των κίβδηλων νομισμάτων καθώς και την ανάπτυξη αλγορίθμων για την επίλυση του προβλήματος των κίβδηλων νομισμάτων με τον ελάχιστο αριθμό ζυγισμάτων.

Στο πρώτο κεφάλαιο καθορίζεται το πρόβλημα και δίνονται κάποιες βασικές έννοιες, οι οποίες θα χρησιμοποιηθούν στα υπόλοιπα κεφάλαια.

Στο δεύτερο κεφάλαιο, εξετάζουμε το πρόβλημα όπου έχουμε ένα σύνολο από n νομίσματα εκ των οποίων m είναι κίβδηλα (ελαφρύτερα). Τα κίβδηλα και τα κανονικά νομίσματα έχουν το ίδιο βάρος αντιστοίχως. Θέλουμε να βρούμε τα κίβδηλα νομίσματα κάνοντας τον ελάχιστο αριθμό ζυγισμάτων.

Στο τρίτο κεφάλαιο, αναπτύσσουμε αλγορίθμους για την επίλυση του προβλήματος για ένα κίβδηλο νόμισμα το οποίο δεν γνωρίζουμε αν είναι ελαφρύτερο ή βαρύτερο από τα κανονικά, και για οποιονδήποτε αριθμό νομισμάτων.

Στο τέταρτο κεφάλαιο, μελετάμε κάποιες διαφορετικές εκδοχές του προβλήματος των κίβδηλων νομισμάτων, δίνοντας κάποια εκτενή παραδείγματα.

Στο πέμπτο κεφάλαιο, ασχολούμαστε με την επίλυση του προβλήματος των κίβδηλων νομισμάτων με τη δημιουργία ομάδων.

Τέλος, στο παράρτημα παρατίθεται πρόγραμμα το οποίο υλοποιεί την περίπτωση κατά την οποία έχουμε n νομίσματα ένα εκ των οποίων είναι κίβδηλο και γνωρίζουμε ότι είναι ελαφρύτερο από τα υπόλοιπα.

Περιεχόμενα

Κεφάλαιο 1

Βασικές Έννοιες

1.1 Το Πρόβλημα των Κίβδηλων Νομισμάτων

Θεωρούμε το πρόβλημα της εύρεσης του ελαχίστου αριθμού ζυγισμάτων που επαρκούν για να προσδιοριστεί το κίβδηλο ή τα κίβδηλα νομίσματα μέσα σε ένα σύνολο n νομισμάτων που έχουν την ίδια εμφάνιση, με δεδομένο ότι έχουμε στη διάθεσή μας μια ζυγαριά.

1.1.1 Ορισμός του Προβλήματος

Θεωρούμε το ακόλουθο πρόβλημα: Θέτουμε $X = \{C_1, C_2, \ldots, Cn\}$ ένα σύνολο από n νομίσματα, της ίδιας εμφάνισης μόνο που m από αυτά είναι βαρύτερα ή ελαφρύτερα από τα υπόλοιπα. Υποθέτουμε ότι τα καλά και τα κίβδηλα νομίσματα έχουν το ίδιο βάρος αντιστοίχως. Με δεδομένο ότι έχουμε μια ζυγαριά, θέλουμε να βρούμε τη βέλτιστη διαδικασία ζυγισμάτων, δηλαδή μια διαδικασία η οποία ελαχιστοποιεί τον μέγιστο αριθμό βημάτων (ζυγισμάτων) που απαιτείται προκειμένου να προσδιοριστεί το κίβδηλο ή τα κίβδηλα νομίσματα. Υποθέτουμε πως η ζυγαριά μας αποκαλύπτει, ποιο από τα υποσύνολα του X είναι βαρύτερο, αλλά όχι κατά πόσο.

Θεωρούμε τα υποσύνολα A και B του X τα οποία αποτελούνται από τον ίδιο αριθμό νομισμάτων. Το βήμα (A,B) θα σημαίνει το ζύγισμα του υποσυνόλου A με το B, από το οποίο μπορούν να προκύψουν τα ακόλουθα αποτελέσματα:

a) Τα δύο σύνολα να ισορροπούν, το οποίο συμβολίζεται ως $A = B$.
b) Τα δύο σύνολα να μην ισορροπούν, το οποίο συμβολίζεται ως $A \neq B$. Επίσης χρησιμοποιούμε τον συμβολισμό $A > B$, $B > A$ όπου το σύμβολο $>$ μεταξύ των δύο συνόλων σημαίνει ότι "είναι βαρύτερο από".

Η πιο απλή περίπτωση του προβλήματος είναι όταν έχουμε $m = 1$. Πολλές μελέτες έχουν γίνει πάνω σε αυτή την περίπτωση. Πιο πολύπλοκες είναι οι περιπτώσεις όπου έχουμε περισσότερα του ενός κίβδηλα νομίσματα ($m = 2, 3, 4, \ldots$) . Κι εδώ έχουν γίνει αρκετές έρευνες προκειμένου να προσδιοριστεί ο ελάχιστος αριθμός ζυγισμάτων για την εύρεση των κίβδηλων νομισμάτων.

1.1.2 Εισαγωγή

Σε κάθε αλγόριθμο που υπολογίζει ποια είναι τα κίβδηλα νομίσματα αντιστοιχεί ένα Δένδρο Απόφασης (Δ.Α.).

Κάθε εσωτερικός κόμβος του δένδρου απόφασης αντιστοιχεί σε ένα ζύγισμα και έχει το πολύ τρία παιδιά, διότι αν ζυγίσουμε δυο σύνολα Α και Β μπορούν να προκύψουν τρία αποτελέσματα:

i. Τα δύο σύνολα να ισορροπούν, Α = Β.
ii. Το σύνολο Α να είναι βαρύτερο από το σύνολο Β, Α > Β.
iii. Το σύνολο Α να είναι ελαφρύτερο από το σύνολο Β, Α < Β.

Αν κάποιος εσωτερικός κόμβος έχει λιγότερα από τρία παιδιά, αυτό σημαίνει ότι η περίπτωση που δεν εμφανίζεται είναι αδύνατη.
Έστω ότι σε ένα σύνολο n νομισμάτων τα m από αυτά είναι κίβδηλα. Κάθε φύλλο του Δ.Α. αντιστοιχεί σε μια m-αδα κίβδηλων νομισμάτων.

Προτάσεις

1.Ο ελάχιστος απαιτούμενος αριθμός ζυγισμάτων για κάθε περίπτωση ισούται με το ύψος του δένδρου.
2. Ο αριθμός των φύλλων του Δ.Α. είναι μεγαλύτερος ή ίσος του συνδυασμού $\binom{n}{m}$.

Από τις προτάσεις 1. και 2. προκύπτουν τα ακόλουθα πορίσματα (συμβολίζουμε με $w_{ελ}$ τον ελάχιστο αριθμό ζυγισμάτων).

Πορίσματα

1. Για τον ελάχιστο αριθμό ζυγισμάτων ισχύει ότι $w_{ελ} \geq \left\lceil \log_3 \binom{n}{m} \right\rceil$.

2. Αν ένα δένδρο απόφασης έχει ύψος ίσο με $\left\lceil \log_3 \binom{n}{m} \right\rceil$ τότε είναι το βέλτιστο δυνατό.

Το ερώτημα που τίθεται εδώ είναι αν υπάρχει Δ.Α. με ύψος ακριβώς $\left\lceil \log_3 \binom{n}{m} \right\rceil$ για κάθε ζεύγος n, m.

1.1.3 Παραλλαγές του Προβλήματος των Κίβδηλων Νομισμάτων

Το πρόβλημα των κίβδηλων νομισμάτων έχει πολλές παραλλαγές. Οι παραλλαγές αυτές σχετίζονται με τον αριθμό των νομισμάτων που είναι κίβδηλα, με το αν το κίβδηλο νόμισμα είναι βαρύτερο ή ελαφρύτερο από τα κανονικά νομίσματα, με το εάν υπάρχει ένα κανονικό νόμισμα που είναι γνωστό εκ των προτέρων κτλ. Ενδεικτικά αναφέρουμε παρακάτω μερικές παραλλαγές του προβλήματος:

- Δεδομένος συνολικός αριθμός νομισμάτων εκ των οποίων ένα είναι βαρύτερο / ελαφρύτερο από τα υπόλοιπα.
- Δεδομένος συνολικός αριθμός νομισμάτων εκ των οποίων ένα είναι κανονικό και γνωστό εκ των προτέρων, γνωρίζουμε ότι υπάρχει ένα κίβδηλο νόμισμα και θέλουμε να βρούμε αν αυτό είναι βαρύτερο ή ελαφρύτερο από τα υπόλοιπα νομίσματα.
- Δεδομένος συνολικός αριθμός νομισμάτων εκ των οποίων ένα είναι κανονικό και γνωστό εκ των προτέρων και θέλουμε να βρούμε αν υπάρχει κίβδηλο,

ποιο είναι αυτό και αν είναι βαρύτερο ή ελαφρύτερο από τα υπόλοιπα νομίσματα.

- Δεδομένος συνολικός αριθμός νομισμάτων εκ των οποίων τα δύο είναι κίβδηλα και γνωρίζουμε ότι το ένα είναι ελαφρύτερο από τα υπόλοιπα και το άλλο βαρύτερο.
- Δεδομένος συνολικός αριθμός νομισμάτων εκ των οποίων τα m είναι κίβδηλα και ελαφρύτερα / βαρύτερα από τα υπόλοιπα.
- Δεδομένος συνολικός αριθμός νομισμάτων εκ των οποίων m είναι κίβδηλα και γνωρίζουμε ότι k από αυτά είναι ελαφρύτερα από τα υπόλοιπα και m – k είναι βαρύτερα από τα υπόλοιπα νομίσματα.
- Δεδομένος συνολικός αριθμός νομισμάτων εκ των οποίων έχουμε l που είναι κανονικά και γνωστά εκ των προτέρων, και θέλουμε να βρούμε αν υπάρχει κίβδηλο, ποιο είναι αυτό και να προσδιορίσουμε αν είναι ελαφρύτερο ή βαρύτερο από τα υπόλοιπα.
- Δεδομένος συνολικός αριθμός νομισμάτων εκ των οποίων m είναι κίβδηλα και η ζυγαριά δείχνει μια φορά λάθος ένδειξη.

Κεφάλαιο 2

Πώς να Βρούμε Πολλά Κίβδηλα Νομίσματα

2.1 Εισαγωγικά

Υποθέτουμε πως έχουμε ένα σύνολο από n νομίσματα εκ των οποίων m είναι κίβδηλα (ελαφρύτερα). Τα κίβδηλα και τα κανονικά νομίσματα έχουν το ίδιο βάρος αντιστοίχως.

Θέλουμε να βρούμε τα κίβδηλα νομίσματα χρησιμοποιώντας μια ζυγαριά ισορροπίας.

Θεώρημα

Για τον ελάχιστο αριθμό ζυγισμάτων $\mu_m(n)$ που απαιτείται για την εύρεση των κίβδηλων νομισμάτων, ισχύει ότι:

$$\left\lceil \log_3 \binom{n}{m} \right\rceil \leq \mu_m(n) \leq \left\lceil \log_3 \binom{n}{m} \right\rceil + 15m.$$

Λήμμα 1

Ας υποθέσουμε ότι έχουμε r σύνολα που το καθένα έχει p νομίσματα. Μπορούμε να προσδιορίσουμε τη λίστα όλων των συνόλων που έχουν μέγιστο βάρος χρησιμοποιώντας το πολύ $r - 1$ ζυγίσματα.

Λήμμα 2

Έστω ότι έχουμε m κίβδηλα νομίσματα σε ένα σύνολο από $n = 4mp + q$ νομίσματα, όπου $0 \leq q \leq 4m - 1$. Τότε μπορούμε να βρούμε 3mp γνήσια νομίσματα χρησιμοποιώντας το πολύ $4m - 1$ ζυγίσματα.
Απόδειξη

Αρχικά θα δείξουμε ότι υπάρχουν περισσότερα από 3mp γνήσια νομίσματα στο σωρό. Τα γνήσια νομίσματα είναι σε πληθάριθμο ίσα με όλα τα νομίσματα αν αφαιρέσουμε από αυτά τα κίβδηλα νομίσματα. Άρα για το πλήmθος των γνήσιων νομισμάτων ισχύει ότι:

$$4mp + q - m \geq 4mp + q - mp \geq 3mp + q \geq 3mp.$$

Επιλέγουμε 4mp νομίσματα τα οποία χωρίζουμε σε 4m σύνολα που το καθένα περιέχει p νομίσματα.

Θα δείξουμε ότι ανάμεσα στα 4m σύνολα υπάρχουν το πολύ m σύνολα τα οποία περιέχουν κίβδηλα νομίσματα, δηλαδή υπάρχουν 3m σύνολα που περιέχουν μόνο γνήσια νομίσματα. Έστω ότι υπάρχουν πάνω από m σύνολα που περιέχουν κίβδηλα νομίσματα. Τότε αφού σε κάθε σύνολο από αυτά πρέπει να περιέχεται τουλάχιστον ένα κίβδηλο, ο αριθμός των κίβδηλων νομισμάτων θα είναι μεγαλύτερος από m, το οποίο είναι άτοπο. Τα σύνολα που αποτελούνται από γνήσια νομίσματα θα έχουν μεγαλύτερο βάρος από τα σύνολα που περιέχουν τουλάχιστον ένα κίβδηλο νόμισμα καθώς τα κίβδηλα νομίσματα είναι ελαφρύτερα από τα κανονικά. Έτσι, κάνοντας χρήση του Λήμματος 1 μπορούμε να βρούμε τα σύνολα που αποτελούνται από γνήσια νομίσματα, ο πληθάριθμος των οποίων ισούται με 3m, κάνοντας 4m – 1 ζυγίσματα.

Ορισμός
Ένα σύνολο νομισμάτων ονομάζεται κίβδηλο αν περιέχει ένα ή περισσότερα κίβδηλα νομίσματα.

Λήμμα 3

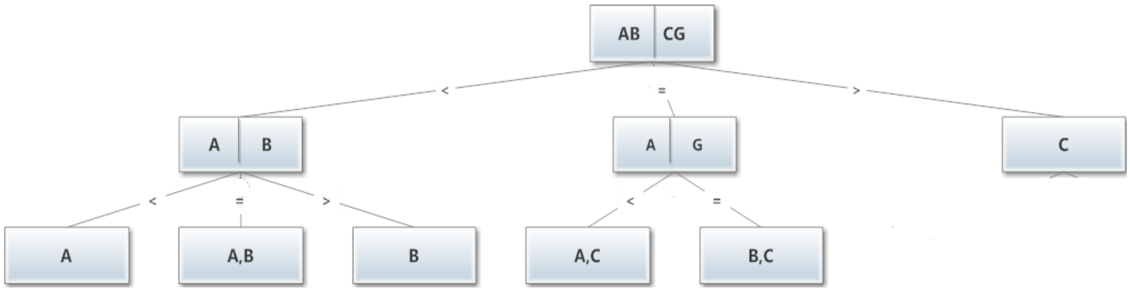
Έστω ότι έχουμε ένα σύνολο G το οποίο περιέχει p γνήσια νομίσματα και ένα σύνολο R που αποτελείται από 3p νομίσματα και περιέχει r κίβδηλα. Χωρίζουμε το σύνολο R σε τρία ίσα μέρη A, B και C. Για να προσδιορίσουμε ποια από τα σύνολα A, B και C είναι κίβδηλα, αν δίνεται ότι:

a) $r = 1$,
b) $1 \le r \le 2$,
c) $r \ge 1$,

χρειαζόμαστε το πολύ 1,2 ή 3 ζυγίσματα αντιστοίχως.
Απόδειξη

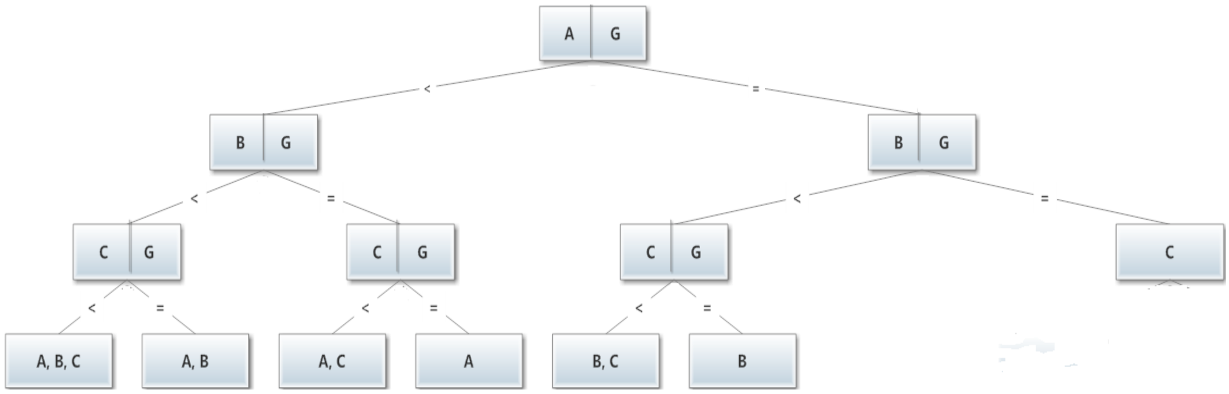
Συμβολίζουμε το βάρος ενός συνόλου X με W(X).

a) Στην περίπτωση όπου έχουμε ένα κίβδηλο νόμισμα ($r = 1$) θα υπάρχει και ένα κίβδηλο σύνολο ανάμεσα στα A, B και C. Έτσι ζυγίζουμε το σύνολο A με το σύνολο B κι έχουμε τρία δυνατά αποτελέσματα: το σύνολο A να είναι βαρύτερο από το σύνολο B ($W(A) > W(B)$), να ισορροπούν ($W(A) = W(B)$) ή το σύνολο B να είναι βαρύτερο από το σύνολο A ($W(A) < W(B)$). Στην περίπτωση που ισχύει $W(A) > W(B)$ το κίβδηλο σύνολο είναι το B. Στην περίπτωση όπου τα δύο σύνολα ισορροπούν ($W(A) = W(B)$) τότε τα σύνολα A και B περιέχουν γνήσια νομίσματα και το σύνολο C είναι το κίβδηλο. Στην τελευταία περίπτωση όπου ισχύει $W(A) < W(B)$ τότε το κίβδηλο σύνολο είναι το A. Συνεπώς χρειαζόμαστε το πολύ ένα ζύγισμα για να βρούμε το αποτέλεσμα.
b) Στην περίπτωση όπου έχουμε $1 \le r \le 2$ έχουμε το ακόλουθο δένδρο απόφασης:



Τα φύλλα του δένδρου αντιστοιχούν στα κίβδηλα σύνολα. Παρατηρούμε ότι χρειαζόμαστε το πολύ 2 ζυγίσματα για να βρούμε τα κίβδηλα σύνολα.

c) Στην περίπτωση όπου έχουμε r ≥ 1 προκύπτει το ακόλουθο δένδρο απόφασης:



Συνεπώς, όπως παρατηρούμε από το παραπάνω δένδρο απόφασης χρειαζόμαστε το πολύ 3 ζυγίσματα για να βρούμε το κίβδηλο ή τα κίβδηλα σύνολα.

2.2 Ενδιάμεσα Αποτελέσματα για την Απόδειξη του Θεωρήματος

Λήμμα 4

Έστω ότι έχουμε x κίβδηλα σύνολα που το καθένα αποτελείται από 3p νομίσματα, που περιέχουν όλα μαζί m κίβδηλα νομίσματα. Υποθέτουμε ακόμα ότι έχουμε ένα επιπλέον σύνολο που αποτελείται από p κανονικά νομίσματα καθώς και δύο λίστες F και G οι οποίες περιέχουν τα σύνολα με το μέγιστο και το δεύτερο μέγιστο βάρος αντίστοιχα. Διαιρούμε κάθε σύνολο σε 3 νέα σύνολα μεγέθους p.

a. Μπορούμε να προσδιορίσουμε ποια από τα καινούρια σύνολα είναι κίβδηλα χρησιμοποιώντας συνολικά το πολύ min(m,3x) ζυγίσματα.
b. Μπορούμε να προσδιορίσουμε τις λίστες με τα νέα σύνολα του μεγίστου και δεύτερου μεγίστου βάρους χρησιμοποιώντας το πολύ 4(y – x) ζυγίσματα, όπου το y δηλώνει τον αριθμό των καινούριων κίβδηλων συνόλων.

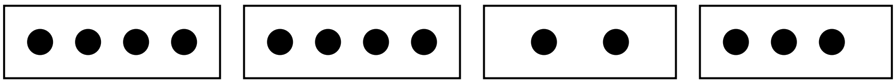
Απόδειξη

Ισχύει ότι $x \leq m$, αφού ο αριθμός των κίβδηλων συνόλων είναι το πολύ ίσος με τον αριθμό των κίβδηλων νομισμάτων. Η ισότητα προκύπτει μόνο όταν σε κάθε κίβδηλο σύνολο έχουμε ένα κίβδηλο νόμισμα.

a) i. Υποθέτουμε ότι $3x \leq m$.

Διαιρούμε καθένα από τα x σύνολα σε 3 σύνολα που περιέχουν p νομίσματα το καθένα. Σύμφωνα με το Λήμμα 3c θα χρειαστούμε το πολύ 3 ζυγίσματα για την εύρεση κάθε κίβδηλου συνόλου. Άρα για τα x κίβδηλα σύνολα θα χρειαστούμε το πολύ $3x \leq m$ ζυγίσματα συνολικά.

Για παράδειγμα, έχουμε 4 κίβδηλα σύνολα ($x = 4$), τα S_1, S_2, S_3 και S_4 τα οποία περιέχουν συνολικά 13 κίβδηλα νομίσματα ($m = 13$). Έχουμε:



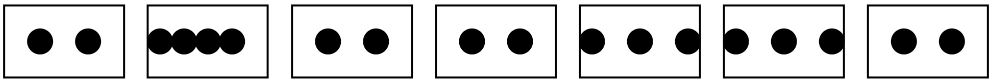
S_1 S_2 S_3 S_4

$3x \leq m \Leftrightarrow 3*4 \leq 13 \Leftrightarrow 12 \leq 13$ ισχύει.

ii. Υποθέτουμε ότι $3x - |F| \leq m < 3x$.

Συνεπώς το F αποτελείται από σύνολα που περιέχουν το πολύ 2 κίβδηλα νομίσματα. Πράγματι, αν κάθε στοιχείο της F περιέχει τουλάχιστον 3 κίβδηλα νομίσματα, τότε $m \geq 3|F| + 4(x - |F|) \Leftrightarrow m \geq 4x - |F| = 3x + (x - |F|) \geq 3x$, άτοπο. Άρα θα εφαρμόσουμε το Λήμμα 3b για να βρούμε τον μέγιστο αριθμό ζυγισμάτων που χρειάζεται για να βρούμε τα νέα κίβδηλα σύνολα. Έτσι για την περίπτωση του F χρειαζόμαστε το πολύ $2|F|$ ζυγίσματα. Για τα υπόλοιπα σύνολα που είναι σε πλήθος ίσα με $x - |F|$ εφαρμόζουμε το Λήμμα 3c και συνεπώς χρειαζόμαστε το πολύ $3(x - |F|)$ ζυγίσματα για να βρούμε τα νέα σύνολα. Έτσι μπορούμε να βρούμε τα καινούρια κίβδηλα σύνολα χρησιμοποιώντας το πολύ $2|F| + 3(X - |F|) \leq m$ ζυγίσματα.

Για παράδειγμα, έχουμε 7 κίβδηλα σύνολα ($x = 7$), τα S_1, S_2, S_3, S4, S_5, S_6 και S_7 τα οποία περιέχουν συνολικά 18 κίβδηλα νομίσματα ($m = 18$). Εδώ η F αποτελείται από σύνολα που περιέχουν το πολύ δύο κίβδηλα νομίσματα. Έχουμε:



S_1 S_2 S_3 S_4 S_5 S_6 S_7

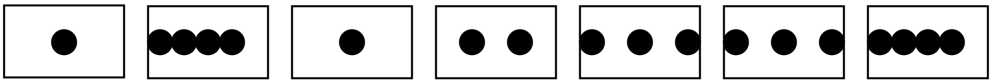
$F = \{S_1, S_3, S_4, S_7\}$, οπότε $|F| = 4$ και

$3x - |F| \leq m < 3x \Leftrightarrow 3*7 - 4 \leq 18 \leq 3*7 \Leftrightarrow 17 \leq 18 < 21$ ισχύει.

iii. Υποθέτουμε ότι $3x - 2|F| \leq m < 3x - |F|$.

Συνεπώς η F αποτελείται από σύνολα που περιέχουν ένα κίβδηλο νόμισμα. Πράγματι, αν κάθε στοιχείο της F περιέχει τουλάχιστον 2 κίβδηλα νομίσματα, τότε $m \geq 2|F| + 3(x - |F|) = 3x - |F|$, άτοπο. Έτσι, εφαρμόζοντας το Λήμμα 3a για τα σύνολα της F, προκύπτει ότι χρειαζόμαστε το πολύ $|F|$ ζυγίσματα για να βρούμε τα καινούρια κίβδηλα σύνολα. Το πλήθος των συνόλων που απομένει είναι ίσο με x - |F|. Σε αυτά εφαρμόζουμε το Λήμμα 3c για να βρούμε τα νέα κίβδηλα σύνολα. Έτσι θέλουμε το πολύ 3(x - |F|) ζυγίσματα για να βρούμε τα νέα κίβδηλα σύνολα. Έτσι θέλουμε συνολικά το πολύ |F| + 3(x - |F|) ≤ m ζυγίσματα.

Για παράδειγμα, έχουμε 7 κίβδηλα σύνολα (x = 7), τα S_1, S_2, S_3, S4, S_5, S_6 και S_7 τα οποία περιέχουν συνολικά 18 κίβδηλα νομίσματα (m = 18). Εδώ η F αποτελείται από σύνολα που περιέχουν ένα κίβδηλο νόμισμα. Έχουμε:

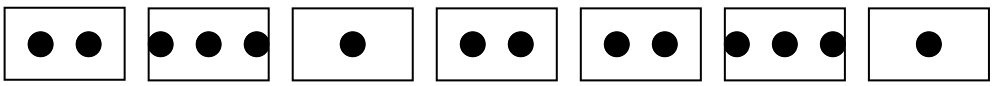


S_1 S_2 S_3 S_4 S_5 S_6 S_7

F = {S_1, S_3}, οπότε $|F| = 2$ *και*

$3x - 2|F| \leq m < 3x - |F| \Leftrightarrow 3*7 – 2*2 \leq 18 < 3*7 – 2 \Leftrightarrow 17 \leq 18 < 19$ *ισχύει.*

iv. Έστω ότι m < 3x - 2|F|.
Συνεπώς η F αποτελείται από σύνολα που περιέχουν ένα κίβδηλο νόμισμα και η G αποτελείται από σύνολα που περιέχουν δύο κίβδηλα νομίσματα. Πράγματι, η F αποτελείται από στοιχεία με ένα κίβδηλο, αφού διαφορετικά, όπως δείξαμε στη (iii) θα είχαμε $m \geq 2|F| + 3 (x - |F|)$ και άρα $m > |F| + 3(x - |F|)$, άτοπο. Επιπλέον, αν κάθε στοιχείο της G περιέχει τουλάχιστον 3 κίβδηλα νομίσματα, τότε $m \geq |F| + 3(x - |F|) = 3x - 2|F|$, άτοπο. Συνεπώς, εφαρμόζουμε το Λήμμα 3a για να βρούμε τα καινούρια κίβδηλα σύνολα που προκύπτουν από την F κι έχουμε το πολύ |F| ζυγίσματα. Για την G, χρησιμοποιώντας το Λήμμα 3b, έχουμε το πολύ 2|G| ζυγίσματα. Ενώ, τέλος, για τα σύνολα που απομένουν, και που είναι σε πλήθος ίσα με x – (|F| + |G|), εφαρμόζοντας το Λήμμα 3c, χρειαζόμαστε το πολύ 3(x – (|F| + |G|)) ζυγίσματα για να βρούμε τα καινούρια κίβδηλα σύνολα. Συνεπώς, έχουμε το πολύ |F| + 2|G| + 3(x – (|F| + |G|)) ≤ m ζυγίσματα και η απόδειξη του (a) είναι πλήρης.
Για παράδειγμα, έχουμε 7 κίβδηλα σύνολα (x = 7), τα S_1, S_2, S_3, S4, S_5, S_6 και S_7 τα οποία περιέχουν συνολικά 14 κίβδηλα νομίσματα (m = 14). Εδώ η F αποτελείται από σύνολα που περιέχουν ένα κίβδηλο νόμισμα και η G αποτελείται από σύνολα που περιέχουν δύο κίβδηλα νομίσματα. Έχουμε:



S_1 S_2 S_3 S_4 S_5 S_6 S_7

F = {S_3, S_7} και G = {S_1, S_4, S_5}, οπότε $|F| = 2$, $|G| = 3$ *και*

$m < 3x - 2|F| \Leftrightarrow 14 < 3*7 – 2*2 \Leftrightarrow 14 < 17$ *ισχύει.*

b) Τα y καινούρια κίβδηλα σύνολα μπορούν να χωριστούν σε δύο ομάδες: την L που είναι η ομάδα των νέων συνόλων που περιέχουν όλα τα κίβδηλα νομίσματα από τα αντίστοιχα παλιά σύνολα, και την H που είναι η ομάδα από τα εναπομείναντα κίβδηλα σύνολα.

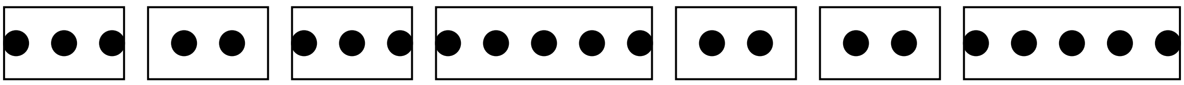
Έστω ότι F΄ και G΄ είναι οι λίστες με τα νέα σύνολα στο L που αντιστοιχούν στα παλιά σύνολα των F και G.

Κάνοντας χρήση του Λήμματος 1 μπορούμε να βρούμε τις λίστες F'' και G'' των νέων συνόλων του μεγίστου και δεύτερου μεγίστου βάρους χρησιμοποιώντας το πολύ $|H|$ - 1 και $|H|$ - 2 ζυγίσματα αντιστοίχως. Έστω ότι $A \in F'$, $B \in G'$, $C \in F''$ και $D \in G''$. Ζυγίζοντας τα ζευγάρια A, C και B, D μπορούμε να βρούμε τα σύνολα του μεγίστου και ελαχίστου βάρους ανάμεσα σε αυτά τα τέσσερα σύνολα και μετά το ζύγισμα των υπόλοιπων δύο μπορούμε να κάνουμε διάταξη αυτών των συνόλων σύμφωνα με το βάρος τους (σε μερικές περιπτώσεις μπορεί να προκύψει ισότητα). Χρησιμοποιήσαμε συνολικά το πολύ $2|H|$ ζυγίσματα. Επιπλέον, τουλάχιστον δύο στοιχεία του H εμφανίζονται σε κάθε μία από τις $(x - |L|)$ τριάδες που περιέχουν στοιχεία του H. Άρα:

$$\left.\begin{array}{l} |H| \geq 2(x - |L|) \Leftrightarrow |H| \geq 2x - 2|L| \\ \text{Αλλά } y = |H| + |L| \text{ και συνεπώς } |L| = y - |H| \end{array}\right\} \Rightarrow$$

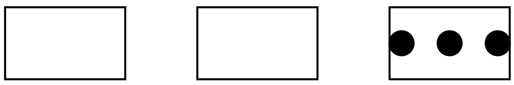
$$|H| \geq 2x - 2(y - |H|) \Leftrightarrow |H| \geq 2x - 2y + 2|H| \Leftrightarrow$$
$$|H| \leq 2\,(y - x).$$

Άρα συνολικά χρησιμοποιήσαμε πράγματι το πολύ $2*2(y - x) = 4(y-x)$ ζυγίσματα.
Για παράδειγμα, έχουμε 7 κίβδηλα σύνολα (x = 7), τα S_1, S_2, S_3, S4, S_5, S_6 και S_7 τα οποία περιέχουν συνολικά 22 κίβδηλα νομίσματα (m = 22). Έχουμε:



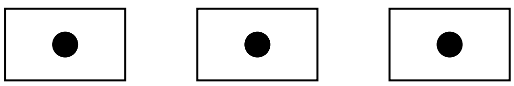
S_1 S_2 S_3 S_4 S_5 S_6 S_7

$F = \{S_2, S_5, S_6\}$ και $G = \{S_1, S_3\}$.
Διαιρούμε καθένα από τα 7 κίβδηλα σύνολα σε 3 υποσύνολα κι έχουμε:

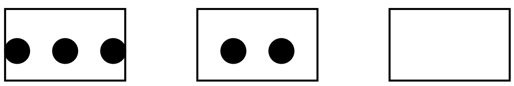


S_{11} S_{12} S_{13}

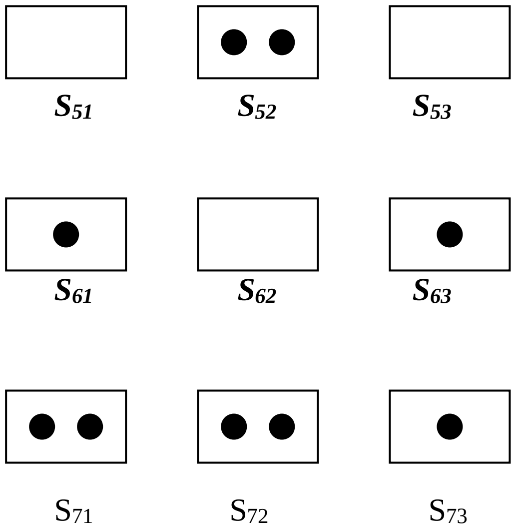
S_{21} S_{22} S_{23}



S_{31} S_{32} S_{33}



S_{41} S_{42} S_{43}



Το πλήθος των καινούριων κίβδηλων συνόλων είναι $y = 13$, η ομάδα των νέων συνόλων που περιέχουν όλα τα κίβδηλα νομίσματα από τα αντίστοιχα παλιά σύνολα είναι η $L = \{S_{13}, S_{21}, S_{52}\}$ και η ομάδα με τα εναπομείναντα κίβδηλα σύνολα είναι η $H = \{S_{31}, S_{32}, S_{33}, S_{41}, S_{42}, S_{61}, S_{63}, S_{71}, S_{72}, S_{73}\}$, με $|H| = 10$. Είναι:

$2(y – x) = 2(13 – 7) = 12 > |H| = 10$.

Έχουμε $F' = \{S_{21}, S_{52}\}$ και $G = \{S_{13}\}$ που είναι οι λίστες με τα νέα σύνολα στο L που αντιστοιχούν στα παλιά σύνολα των F και G. Επίσης, μπορούμε να βρούμε τις λίστες F'' και G'' των νέων συνόλων του μεγίστου και του δεύτερου μεγίστου βάρους χρησιμοποιώντας το πολύ $|H| - 1 = 9$ και $|H| - 2 = 8$ ζυγίσματα αντιστοίχως. Είναι $F'' = \{S_{31}, S_{32}, S_{33}, S_{61}, S_{63}, S_{73}\}$ και $G'' = \{S_{42}, S_{71}, S_{72}\}$. Έστω ότι $A = S_{21} \in F'$, $B = S_{13} \in G'$, $C = S_{31} \in F''$ και $D = S_{42} \in G''$. Ζυγίζοντας τα ζευγάρια A, C και B, D μπορούμε να βρούμε τα σύνολα του μεγίστου και ελαχίστου βάρους ανάμεσα σε αυτά τα τέσσερα σύνολα και μετά το ζύγισμα των υπόλοιπων δύο μπορούμε να κάνουμε διάταξη αυτών των συνόλων σύμφωνα με το βάρος τους. Είναι $C > A$, $D > B$ και $A = D$ και τελικά προκύπτει η κατάταξη $C > A = D > B$. Συνολικά χρειαζόμαστε το πολύ $(|H| - 1) + (|H| -2) + 2 + 1 = 2|H| = 20$ ζυγίσματα και αφού $|H| = 10 \leq 12 = 2(y – x)$, χρειαζόμαστε τελικά το πολύ $2*2(y – x) = 4(y – x) = 24$ ζυγίσματα.

Λήμμα 5

Έστω ότι έχουμε $n = 3^k$ νομίσματα κι έναν γνωστό αριθμό m κίβδηλων νομισμάτων ανάμεσα σε αυτά, και επιπλέον 3^{k-1} κανονικά νομίσματα. Τότε τα κίβδηλα νομίσματα μπορούν να προσδιοριστούν χρησιμοποιώντας το πολύ $(k – [\log_3 m] + 6)\, m$ ζυγίσματα.

Απόδειξη

Ο αλγόριθμός μας αποτελείται από k βήματα. Μετά από το βήμα i έχουμε $x(i) \leq 3^i$ κίβδηλα σύνολα ($x(0) = 1$), και δυο λίστες F(i) και G(i) που αποτελούνται από τα σύνολα με το μέγιστο και δεύτερο μέγιστο βάρος. Διαιρούμε κάθε σύνολο σε τρία ίσα μέρη, προσδιορίζουμε ποια από τα νέα σύνολα είναι κίβδηλα και καθορίζουμε τις νέες λίστες F(i + 1) και G(i + 1) που αποτελούνται από τα σύνολα με το μέγιστο βάρος και το δεύτερο μέγιστο βάρος αντιστοίχως.

Στο πρώτο βήμα έχουμε ένα κίβδηλο σύνολο $x = 1$ και $y \leq 3$ (όπου y είναι ο αριθμός των καινούριων κίβδηλων συνόλων) και τα νέα κίβδηλα σύνολα τα βρίσκουμε μετά από min (m, 3*1) ζυγίσματα. Έπειτα έχουμε $x = 3$ κίβδηλα σύνολα και για να βρούμε τα νέα κίβδηλα σύνολα που θα προκύψουν χρειαζόμαστε min (m, 3*3) ζυγίσματα. Συνεχίζοντας επαγωγικά την απόδειξή μας, παρατηρούμε ότι από τις k φορές, τις t φορές θα πρέπει να κάνουμε 3^i ζυγίσματα για να βρούμε τα νέα κίβδηλα σύνολα, ενώ τις υπόλοιπες k-t φορές θα κάνουμε m ζυγίσματα για να βρούμε τα νέα κίβδηλα σύνολα. Έτσι έχουμε:

$$(3 + \ldots + 3^t) + (k - t)m + \sum_{i=0}^{k-1} 4(x(i+1) - x(i)) \text{ ζυγίσματα}$$

όπου $t = [\log_3 m]$.

Έχουμε όμως

$$\sum_{i=0}^{k-1} 4(x(i+1) - x(i)) = 4(x(1) - x(0)) + 4(x(2) - x(1)) + \ldots + 4(x(k) - x(k-1)) = 4(m - 1)$$

και

$$3 + \ldots + 3^t = 3\frac{3^t - 1}{3 - 1} = 3\frac{m - 1}{2}.$$

Είναι

$$3\frac{m-1}{2} + [k - \log_3 m]m + 4(m - 1) = \frac{3}{2}m - \frac{3}{2} + (k - [\log_3 m])m + 4m - 4 \leq 6m + km - [\log_3 m]m$$

κι έτσι αποδεικνύεται η παραπάνω πρόταση.

Για να μπορέσουμε να ολοκληρώσουμε την απόδειξη του Θεωρήματος χρειαζόμαστε μερικούς ορισμούς κι ένα ακόμη αποτέλεσμα από τη Θεωρία Γραφημάτων.

1. **Δένδρο με ρίζα** λέγεται ένα δένδρο, στο οποίο ένας συγκεκριμένος κόμβος του χρησιμοποιείται ως η ρίζα του.
2. **Φύλλο** ενός δένδρου με ρίζα λέγεται κάθε κόμβος που δεν έχει παιδιά.
3. **Ύψος** ενός δένδρου με ρίζα λέγεται ο αριθμός των κόμβων που περιέχονται με ένα μονοπάτι μέγιστου μήκους από τη ρίζα ως ένα κόμβο του δένδρου (ο οποίος προφανώς πρέπει να είναι φύλλο).
4. Κάθε κόμβος του δένδρου που δεν είναι φύλλο, ονομάζεται **εσωτερικός κόμβος.**
5. **Κ-δένδρο** ονομάζεται ένα δένδρο με ρίζα, όταν κάθε κόμβος του επιτρέπεται να έχει το πολύ k παιδιά.
6. **Δένδρο απόφασης** λέγεται ένα k-δένδρο του οποίου κάθε εσωτερικός κόμβος παριστάνει μια ερώτηση για την οποία πρέπει να αποφασίσουμε. Οι δυνατές απαντήσεις (αποφάσεις) στην ερώτηση παριστάνονται από τους (k το πολύ) δεσμούς που συνδέουν τον κόμβο με τα παιδιά του. Τα τελικά αποτελέσματα (αποφάσεις) της διαδικασίας παριστάνονται από τα φύλλα του δένδρου.

Λήμμα 6

Το ύψος h ενός k-δένδρου με l φύλλα είναι τουλάχιστον $\log_k l+1$.

Απόδειξη:

Υπάρχουν $\binom{n}{m}$ πιθανές τελικές απαντήσεις (φύλλα) στο δένδρο απόφασης.

Το δένδρο απόφασης εδώ είναι ένα 3-δένδρο, αφού κάθε φορά που ζυγίζουμε υπάρχουν 3 δυνατά αποτελέσματα:

$<$: το αριστερό είναι ελαφρύτερο,

$=$: τα δύο μέρη έχουν το ίδιο βάρος,

$>$: το αριστερό μέρος είναι βαρύτερο.

Άρα το ύψος του δένδρου είναι τουλάχιστον $\log_3 \binom{n}{m} + 1$, οπότε το πλήθος των δεσμών του δένδρου (δηλαδή των ζυγισμάτων) από τη ρίζα μέχρι κάποιο (τουλάχιστον ένα) φύλλο (δηλαδή σε μία τουλάχιστον περίπτωση) είναι τουλάχιστον $\log_3 \binom{n}{m}$.

2.3 Το Τέλος της Απόδειξης

Η πρώτη ανισότητα του Θεωρήματος, προκύπτει από το Λήμμα 6. Για την δεύτερη ανισότητα υποθέτουμε ότι έχουμε $n = 4mp + q$ νομίσματα, όπου $0 \leq q \leq 4m - 1$. Από το Λήμμα 2 μπορούμε να βρούμε 3mp νομίσματα που είναι γνήσια, χρησιμοποιώντας το πολύ $4m - 1$ ζυγίσματα. Ζυγίζοντας ένα από τα καλά νομίσματα με q από τα εναπομείναντα τελειώνουμε με mp νομίσματα που περιέχουν έναν δοθέντα αριθμό $m_0 \leq m$ κίβδηλα. Έχουμε ήδη βρει τα $m - m_0$ από τα κίβδηλα νομίσματα. Υποθέτουμε ότι $n_0 = 3^k$ είναι η μικρότερη δύναμη για την οποία ισχύει $3^k \geq mp$. Έπειτα, προσθέτοντας $3^k - mp \leq 2mp$ από τα γνήσια νομίσματα στο εναπομείναν σύνολο, παίρνουμε ένα επιπλέον σύνολο από $mp \geq 3^{k-1}$ γνήσια νομίσματα.

Είναι:

$3^{k-1} \leq mp$ (αφού η k είναι η ελάχιστη δύναμη για την οποία ισχύει η ανισότητα $3^{k-1} \geq mp$). Άρα $3^k \leq 3mp$, οπότε $3^k - mp \leq 2mp$.

Εφαρμόζοντας λοιπόν το Λήμμα 5, μπορούμε να βρούμε τα m_0 κίβδηλα νομίσματα χρησιμοποιώντας $(k - [\log_3 m_0] + 6)m_0$ ζυγίσματα. Είναι ξεκάθαρο ότι χρησιμοποιούμε συνολικά το πολύ $([\log_3 n] - [\log_3 m] +14)m$ ζυγίσματα. Έχουμε επίσης ότι $\mu_m(x) \leq ([\log_3 n] - [\log_3 m] + 14)m \leq (\log_3 n - \log_3 m + 15)m = (\log_3 n - \log_3 m)m + 15m = \log_3 (n/m)^m + 15m \leq \log_3 \binom{n}{m} + 15m$.

Παρατήρηση

Κατά τη διάρκεια ανάπτυξης του αλγορίθμου χρησιμοποιήθηκε η πληροφορία ότι έχουμε λιγότερα ή ίσα από m κίβδηλα νομίσματα. Από την άλλη πλευρά ο αλγόριθμος δε θα δούλευε αν δεν είχαμε αυτή την πληροφορία.

Κεφάλαιο 3

Ένας Γενικευμένος Αλγόριθμος για να Λύσουμε το Πρόβλημα των n Νομισμάτων με ένα Κίβδηλο Νόμισμα

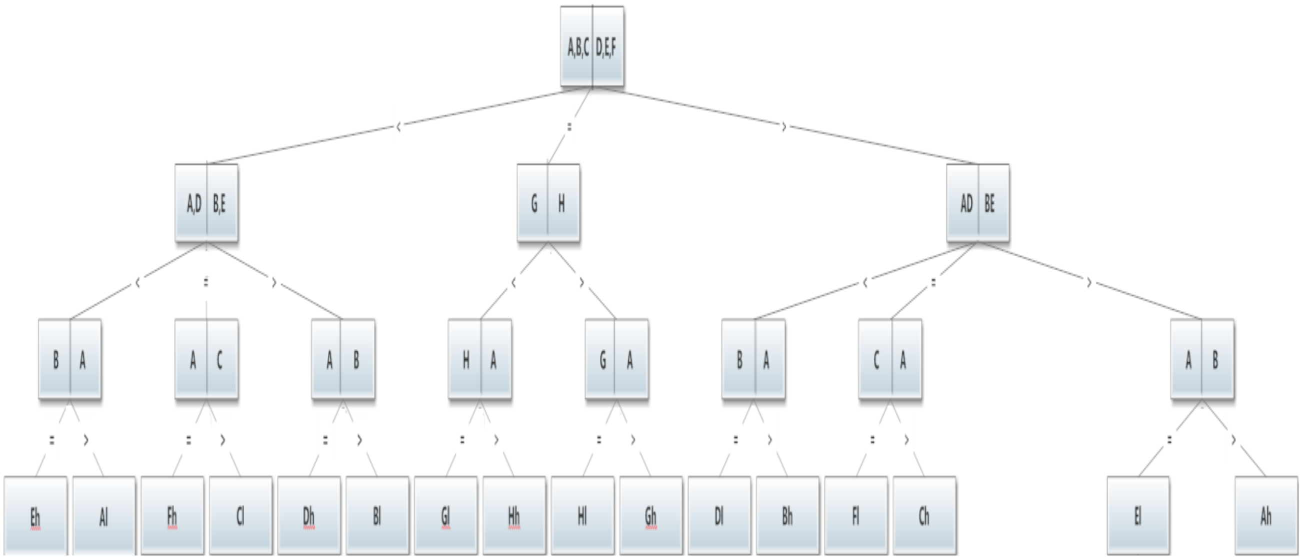
3.1 Εισαγωγή

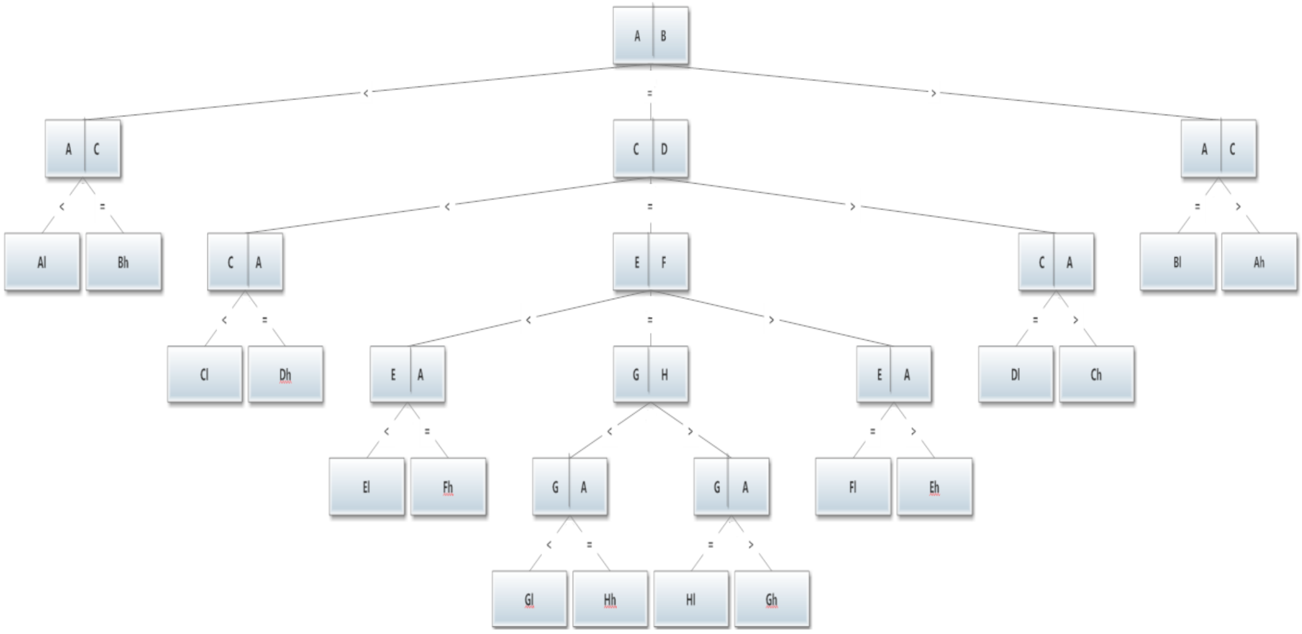
Εδώ θα αναπτύξουμε αλγορίθμους για την επίλυση του προβλήματος των κίβδηλων νομισμάτων για οποιονδήποτε αριθμό νομισμάτων (n). Ο πρώτος αλγόριθμος βασίζεται στην υπάρχουσα κλασική λύση για το πρόβλημα των οκτώ νομισμάτων (με μια μικρή τροποποίηση) για μεγαλύτερες τιμές του n, όπου το n είναι μια δύναμη του 2. Έπειτα αναπτύσσεται ένας αλγόριθμος για την επίλυση του προβλήματος των n νομισμάτων, όπου n είναι ζυγός αλλά όχι δύναμη του 2, π.χ. 6, 10, 12, 14, 18, 20 κλπ. Στο τέλος ο ίδιος αλγόριθμος επεκτείνεται για την επίλυση του προβλήματος των κίβδηλων νομισμάτων για περιττό αριθμό νομισμάτων.

Πρόβλημα των 8 νομισμάτων
Δίνονται 8 νομίσματα: A, B, C, D, E, F, G και H.
Υπάρχει ένα κίβδηλο ανάμεσά τους το οποίο δεν γνωρίζουμε αν είναι βαρύτερο ή ελαφρύτερο από τα υπόλοιπα.
Στόχος μας είναι να βρούμε το κίβδηλο νόμισμα, με τον ελάχιστο δυνατό αριθμό ζυγισμάτων.

3.2 Βιβλιογραφική Έρευνα

Τα παρακάτω σχήματα δείχνουν τις κλασικές λύσεις που βρέθηκαν για το πρόβλημα των 8 νομισμάτων. Στην πρώτη εικόνα το πρόβλημα λύνεται με τρία μόνο ζυγίσματα, ενώ στην δεύτερη εικόνα ο αριθμός ζυγισμάτων αυξάνεται σε πέντε. Στον αλγόριθμο που αναπτύσσεται παρακάτω στόχος μας είναι να βρούμε το κίβδηλο νόμισμα με τον ελάχιστο δυνατό αριθμό ζυγισμάτων.





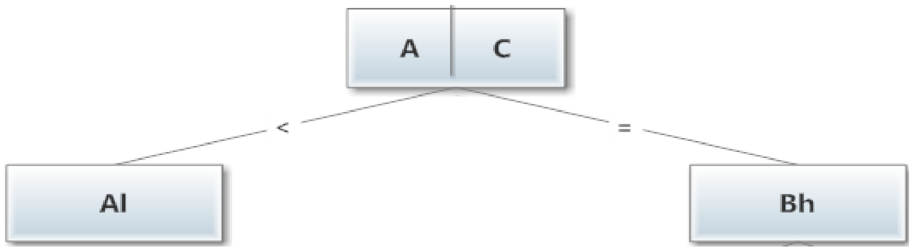
Οι δείκτες h (heavier) και l (lighter) χρησιμοποιούνται για να δηλώσουν αν το κέρμα είναι βαρύτερο ή ελαφρύτερο αντίστοιχα.

3.3 Αλγόριθμοι για την Επίλυση του Προβλήματος των n Νομισμάτων

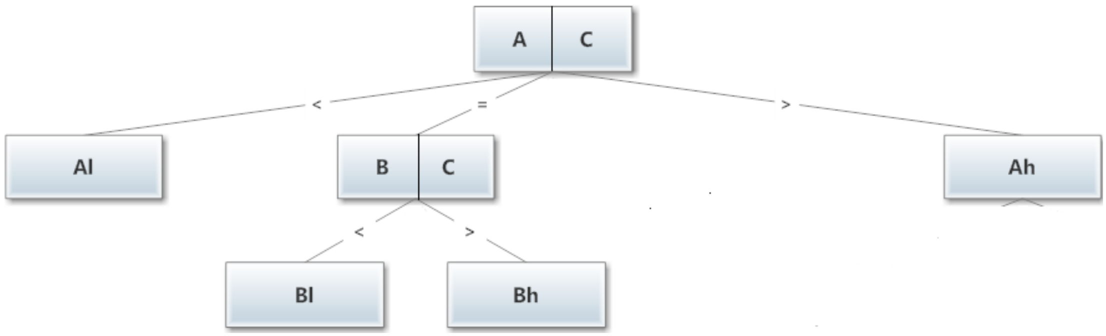
3.3.1 Ένας αλγόριθμος για την επίλυση του προβλήματος των n νομισμάτων όπου n δύναμη του 2, με n ≥ 8

Πριν αναπτύξουμε τον γενικό αλγόριθμο για τη δύναμη του 2 (ο οποίος αναφέρεται σε n ≥ 8), θα παρουσιάσουμε κάποιες ειδικές περιπτώσεις. Συγκεκριμένα, και δεδομένου ότι προφανώς το πρόβλημα δε μπορεί να λυθεί για n = 2, οι ειδικές περιπτώσεις που θα μας απασχολήσουν αρχικά είναι οι εξής:

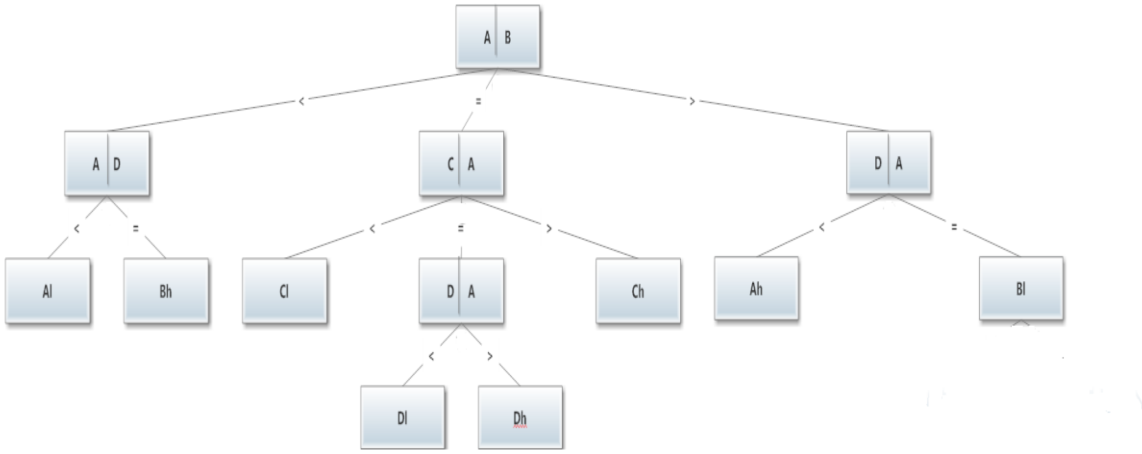
i. Έχουμε δύο νομίσματα A, B από τα οποία γνωρίζουμε ότι ένα είναι κίβδηλο, και επιπλέον γνωρίζουμε ότι το A δεν είναι βαρύ (δηλαδή είναι ελαφρύ, ή κανονικό) και το B δεν είναι ελαφρύ (δηλαδή είναι βαρύ, ή κανονικό). Έχουμε επιπλέον ένα κανονικό νόμισμα C. Τότε εφαρμόζουμε τη “διαδικασία ελέγχου” (check procedure):



ii. Έχουμε δυο νομίσματα A, B από τα οποία γνωρίζουμε ότι ένα είναι κίβδηλο, και έχουμε επιπλέον ένα κανονικό νόμισμα C. Τότε εφαρμόζουμε τη διαδικασία του προβλήματος των 2-νομισμάτων (2-coin problem procedure):



Επίσης, χωριστά πρέπει να αντιμετωπίσουμε και την περίπτωση n = 4, (δηλαδή έχουμε τέσσερα νομίσματα A, B, C, D από τα οποία γνωρίζουμε ότι ένα είναι κίβδηλο). Τότε εφαρμόζουμε τη διαδικασία του προβλήματος των 4-νομισμάτων (4-coin problem procedure):



Το πρόβλημα των n νομισμάτων όπου n δύναμη του 2, με n ≥ 8 λύνεται χρησιμοποιώντας τη διαδικασία n_coin_problem όπως περιγράφεται σε αυτήν την ενότητα. Αυτή η διαδικασία παίρνει σαν εισόδους, τον αρχικό δείκτη των n νομισμάτων και τον αριθμό των νομισμάτων. Αυτή η διαδικασία στη συνέχεια καλεί τη συνάρτηση less_than ή τη συνάρτηση greater_than ανάλογα με το βάρος της ζυγαριάς. Η συνάρτηση less_than καλείται όταν στην πρώτη σύγκριση (η οποία γίνεται στη ρίζα του δένδρου απόφασης) το βάρος του αριστερού μέρους της ζυγαριάς είναι μικρότερο από το βάρος του δεξιού μέρους της ζυγαριάς. Παρόμοια, η συνάρτηση greater_than καλείται όταν το αριστερό μέρος της ζυγαριάς είναι βαρύτερο από το δεξί. Αυτές οι συναρτήσεις καλούνται αναδρομικά.

Καθώς αναπτύσσεται η δομή του δένδρου, κάθε μια από αυτές τις διαδικασίες μειώνει τον αριθμό των νομισμάτων σε κάθε βήμα, μέχρι ο αριθμός των νομισμάτων να γίνει 2. Σε αυτό το βήμα αυτές οι συναρτήσεις παίρνουν τη βοήθεια της συνάρτησης ελέγχου (check function). Όταν η διαδικασία n_coin_problem καλείται αναδρομικά, και ο αριθμός των νομισμάτων που έχουν μείνει είναι μόνο 2 ή μόνο 4, τότε η διαδικασία καλεί τη 2_coin_problem ή τη 4_coin_problem αντιστοίχως (αντί να καλέσει ξανά την n_coin_problem). Τώρα, θα δούμε τη λειτουργία του γενικευμένου αλγορίθμου για την επίλυση του προβλήματος των n νομισμάτων, όπου το n είναι μια δύναμη του 2 μεγαλύτερη ή ίση του 8.
<u>Είσοδος:</u> Ένας ακέραιος αριθμός n νομισμάτων (από τα οποία ένα μόνο νόμισμα είναι κίβδηλο, είτε βαρύτερο είτε ελαφρύτερο από τα υπόλοιπα).
<u>Έξοδος:</u> Ο δείκτης z του κίβδηλου νομίσματος, όπου ο z είναι ακέραιος (θέση του κίβδηλου νομίσματος), και δηλώνει ότι είναι βαρύτερο ή ελαφρύτερο από τα υπόλοιπα.

Η Διαδικασία n_coin_problem

Μας δίνονται n νομίσματα, από τα οποία ένα μόνο νόμισμα είναι κίβδηλο. Για να βρούμε το κίβδηλο νόμισμα (και να προσδιορίσουμε επίσης αν είναι βαρύτερο ή ελαφρύτερο από τα υπόλοιπα), χρησιμοποιούμε τη δομή του δένδρου απόφασης. Καλούμε το n_coins_problem με *αρχή = 1* και *αριθμό νομισμάτων = n*. Εδώ η αρχή δηλώνει τον αρχικό δείκτη των νομισμάτων που θα συγκριθούν, και n είναι το μέγεθος του προβλήματος. Ο συνολικός αριθμός των πιθανών αποτελεσμάτων στην περίπτωση του προβλήματος των n νομισμάτων ισούται με 2n. Αρχικά προσδιορίζουμε τον αριθμό νομισμάτων που βάζουμε στο κάθε μέρος της ζυγαριάς. Αυτός ο αριθμός ισούται με 3x, όπου x = n/8. Έτσι, αρχικά συγκρίνονται 6x

νομίσματα και αφήνουμε έξω τα τελευταία 2x νομίσματα. Για παράδειγμα στην περίπτωση που έχουμε n = 8 τότε προφανώς είναι x = 1 και ο αριθμός των νομισμάτων που συγκρίνονται στη ρίζα του δένδρου απόφασης είναι 6, θέτοντας τα πρώτα 3 νομίσματα στο αριστερό μέρος της ζυγαριάς και τα επόμενα 3 στο δεξί. Παρόμοια, για n = 16 έχουμε x = 2 και ο αριθμός νομισμάτων που συγκρίνονται αρχικά είναι 12, θέτοντας τα πρώτα 6 νομίσματα στο αριστερό μέρος της ζυγαριάς και τα υπόλοιπα 6 στο δεξί μέρος της ζυγαριάς κοκ.
Στο παράδειγμα που ακολουθεί έχουμε n = 32 νομίσματα. Σε αυτή την περίπτωση έχουμε x = 32/8 = 4 και ο αριθμός των νομισμάτων που συγκρίνονται αρχικά ισούται με 24, θέτοντας τα πρώτα 12 νομίσματα στο αριστερό μέρος της ζυγαριάς και τα υπόλοιπα 12 στο δεξί.
Μπορούμε να έχουμε τις ακόλουθες τρεις περιπτώσεις:

1. Το αριστερό μέρος της ζυγαριάς είναι ελαφρύτερο από το δεξί. Αυτή την περίπτωση τη διαχειρίζεται η συνάρτηση less_than. Εφόσον το αριστερό μέρος της ζυγαριάς είναι ελαφρύτερο, αυτό σημαίνει ότι κάποιο από τα 3x νομίσματα που βρίσκεται στο αριστερό μέρος της ζυγαριάς είναι ελαφρύτερο, ή κάποιο από τα 3x νομίσματα που βρίσκονται στο δεξί μέρος της ζυγαριάς είναι βαρύτερο. Οπότε έχουμε 2*3x = 6x πιθανά αποτελέσματα (φύλλα). Το κίβδηλο νόμισμα υπάρχει ανάμεσα στα νομίσματα που δεικτοδοτήθηκαν από την αρχή ως το 6x. Το αριστερό μέρος της ζυγαριάς περιέχει νομίσματα που δεικτοδοτήθηκαν από την αρχή ως το 3x, και το δεξί μέρος περιέχει τα νομίσματα που δεικτοδοτήθηκαν από το 3x+1 ως το 6x. Το πρώτο νόμισμα στο αριστερό μέρος της ζυγαριάς δεικτοδοτείται ως S_left. Παρόμοια το πρώτο νόμισμα στο δεξί μέρος της ζυγαριάς δεικτοδοτείται ως S_right.
Στο παράδειγμα μας, κάποιο από τα νομίσματα 1,2,…,12 που βρίσκεται στο αριστερό μέρος της ζυγαριάς είναι ελαφρύτερο ή κάποιο από τα νομίσματα 13,14,…,24 που βρίσκεται στο δεξί μέρος της ζυγαριάς είναι βαρύτερο. Συνεπώς έχουμε 24 πιθανά αποτελέσματα (φύλλα). Το S_left αρχικά είναι το 1 και το S_right είναι το 13.
2. Το αριστερό μέρος της ζυγαριάς είναι βαρύτερο από το δεξί. Αυτή την περίπτωση τη διαχειρίζεται η συνάρτηση greater_than. Εφόσον το δεξί μέρος της ζυγαριάς είναι ελαφρύτερο, αυτό σημαίνει ότι κάποιο από τα 3x νομίσματα που βρίσκονται στο αριστερό μέρος της ζυγαριάς είναι βαρύτερο ή κάποιο από τα 3x νομίσματα που βρίσκονται στο δεξί μέρος είναι ελαφρύτερο. Παρόμοια, έχουμε 2*3x πιθανά αποτελέσματα (φύλλα). Η δεικτοδότηση είναι ίδια με την προηγούμενη περίπτωση.
Στο παράδειγμα μας κάποιο από τα νομίσματα 1,2,…,12 που βρίσκονται στο αριστερό μέρος της ζυγαριάς είναι βαρύτερο ή κάποιο από τα νομίσματα 13,14,…,24 που βρίσκονται στο δεξί μέρος της ζυγαριάς είναι ελαφρύτερο. Κι εδώ έχουμε 24 πιθανά αποτελέσματα. Και σε αυτή την περίπτωση έχουμε S_left = 1 και S_Right = 13.
3. Τα δύο μέρη της ζυγαριάς ισορροπούν. Εφόσον η ζυγαριά ισορροπεί, αυτό σημαίνει ότι τα 6x νομίσματα τα οποία συγκρίνονται στη ρίζα είναι κανονικά. Συνεπώς το κίβδηλο νόμισμα βρίσκεται στα 2x νομίσματα τα οποία δεν έχουν ζυγιστεί ακόμη. Σε αυτή την περίπτωση, δηλαδή στην περίπτωση της ισότητας, το πρόβλημα των n νομισμάτων υποβαθμίζεται σε πρόβλημα των 2x νομισμάτων. Για να λύσουμε αυτό το πρόβλημα των 2x νομισμάτων, καλούμε αναδρομικά την n_coin_problem, με τον αριθμό των νομισμάτων να ισούται σε αυτή την περίπτωση με 2x. Εφόσον τα πρώτα 6x νομίσματα είναι

κανονικά, έχουμε *αρχή = 6x + 1* και *n = 2x*. Από την περίπτωση της ισότητας μπορούμε να έχουμε 2*2x = 4x πιθανά αποτελέσματα (φύλλα).
Στο παράδειγμά μας, τα νομίσματα 1,2,3,…,23,24 είναι κανονικά, οπότε το κίβδηλο βρίσκεται στα 8 νομίσματα που δεν έχουν ζυγιστεί ακόμη (25,26,…,32). Επομένως έχουμε αρχή = 25 και n=8.

Η Διαδικασία less_than

Σε αυτή τη διαδικασία, επιλέγουμε 4x = 4(n/8) = n/2 από τα 6x νομίσματα ως εξής: διατηρούμε τα πρώτα x (x = n/8) νομίσματα (αρχίζοντας από το S_left) και ανταλλάσσουμε τα επόμενα x νομίσματα με τα πρώτα x νομίσματα που υπάρχουν στο δεξί μέρος της ζυγαριάς (αρχίζοντας από το S_right), και διατηρούμε τα επόμενα x νομίσματα του δεξιού μέρους της ζυγαριάς όπως ήταν. Έτσι, σε αυτή τη σύγκριση, τα τελευταία x νομίσματα και από τα δύο μέρη της ζυγαριάς δε λαμβάνονται υπόψη. Συνεχίζουμε (με n/2 αντί για n).
Στο παράδειγμα μας, επιλέγουμε n/2 = 16 από τα 24 νομίσματα ως εξής: διατηρούμε τα πρώτα 4 νομίσματα (1,2,3,4) στο αριστερό μέρος της ζυγαριάς και ανταλλάσσουμε τα επόμενα 4 νομίσματα (5,6,7,8) με τα πρώτα 4 νομίσματα που υπάρχουν στο δεξί μέρος της ζυγαριάς (13,14,15,16), και διατηρούμε τα επόμενα 4 νομίσματα του δεξιού μέρους της ζυγαριάς (17,18,19,29) όπως ήταν. Έτσι, σε αυτή τη σύγκριση, τα τελευταία 4 νομίσματα από κάθε μέρος της ζυγαριάς δε λαμβάνονται υπόψη (δηλαδή τα νομίσματα 9,10,11,12 του αριστερού μέρους της ζυγαριάς και τα νομίσματα 21,22,23,24 του δεξιού μέρους της ζυγαριάς).
Και σε αυτή τη περίπτωση υπάρχουν τρία δυνατά αποτελέσματα:

1. *Το αριστερό μέρος είναι ελαφρύτερο:* Το κίβδηλο νόμισμα βρίσκεται ανάμεσα στα πρώτα x νομίσματα του αριστερού μέρους της ζυγαριάς, ή στα τελευταία x νομίσματα του δεξιού μέρους της ζυγαριάς. Διαιρούμε τα νομίσματα που βρίσκονται σε κάθε ένα από τα δύο μέρη της ζυγαριάς σε τέσσερα τμήματα: τα s_{l1}, s_{l2}, s_{l3} και s_{l4} για το αριστερό μέρος της ζυγαριάς, και τα s_{r1}, s_{r2}, s_{r3} και s_{r4} για το δεξί μέρος της ζυγαριάς, καθένα από τα οποία περιέχει τον ίδιο αριθμό νομισμάτων. Έπειτα διαγράφουμε τα νομίσματα που βρίσκονται στα δύο τελευταία τμήματα του αριστερού μέρους της ζυγαριάς (δηλαδή τα s_{l3} και s_{l4}) και αυτά που βρίσκονται στα δύο πρώτα τμήματα του δεξιού μέρους της ζυγαριάς (δηλαδή τα s_{r1} και s_{r2}). Το S_left είναι τώρα το πρώτο από τα εναπομείναντα νομίσματα του αριστερού μέρους της ζυγαριάς και το S_right είναι το πρώτο από τα εναπομείναντα νομίσματα του δεξιού μέρους της ζυγαριάς. Τέλος, διατηρούμε τα νομίσματα του s_{l1} του αριστερού μέρους της ζυγαριάς και ανταλλάσσουμε τα νομίσματα του s_{l2} με τα νομίσματα του s_{r3} του δεξιού μέρους της ζυγαριάς, και διατηρούμε τα νομίσματα του s_{r4} ως έχουν.
   Στο παράδειγμά μας, το κίβδηλο βρίσκεται ανάμεσα στα πρώτα 4 νομίσματα του αριστερού μέρους (1,2,3,4) ή στα τελευταία 4 νομίσματα του δεξιού μέρους της ζυγαριάς (17,18,19,20). Συνεπώς διαγράφουμε τα νομίσματα (13,14,15,16) του αριστερού μέρους της ζυγαριάς και τα νομίσματα (5,6,7,8) του δεξιού μέρους της ζυγαριάς. Έτσι έχουμε S_left =1 και S_right = 17.
2. *Το αριστερό μέρος είναι βαρύτερο:* Το κίβδηλο νόμισμα βρίσκεται ανάμεσα στα τελευταία x νομίσματα στο αριστερό μέρος της ζυγαριάς, ή στα πρώτα x νομίσματα στο δεξί μέρος της ζυγαριάς. Διαιρούμε τα νομίσματα που

βρίσκονται σε κάθε ένα από τα δύο μέρη της ζυγαριάς σε τέσσερα τμήματα: τα s_{l1}, s_{l2}, s_{l3} και s_{l4} για το αριστερό μέρος της ζυγαριάς, και τα s_{r1}, s_{r2}, s_{r3} και s_{r4} για το δεξί μέρος της ζυγαριάς, καθένα από τα οποία περιέχει τον ίδιο αριθμό νομισμάτων. Έπειτα διαγράφουμε τα νομίσματα που βρίσκονται στα δύο πρώτα τμήματα του αριστερού μέρους της ζυγαριάς (δηλαδή τα s_{l1} και s_{l2}) και αυτά που βρίσκονται στα δύο τελευταία τμήματα του δεξιού μέρους της ζυγαριάς (δηλαδή τα s_{r3} και s_{r4}). Το S_left είναι τώρα το πρώτο από τα εναπομείναντα νομίσματα του αριστερού μέρους της ζυγαριάς και το S_right είναι το πρώτο από τα εναπομείναντα νομίσματα του δεξιού μέρους της ζυγαριάς. Τέλος, ανταλλάσσουμε τα νομίσματα του s_{l3} του αριστερού μέρους της ζυγαριάς με τα νομίσματα του s_{r2} του δεξιού μέρους της ζυγαριάς, και διατηρούμε τα νομίσματα του s_{l4} και του s_{r1} ως έχουν.
Στο παράδειγμά μας, το κίβδηλο βρίσκεται ανάμεσα στα τελευταία 4 νομίσματα του αριστερού μέρους της ζυγαριάς (13,14,15,16), ή στα πρώτα 4 νομίσματα του δεξιού μέρους της ζυγαριάς (5,6,7,8). Οπότε πρέπει να ανανεώσουμε την τιμή του S_left. Συνεπώς διαγράφουμε τα νομίσματα (1,2,3,4) του αριστερού μέρους της ζυγαριάς και τα νομίσματα (17,18,19,20) του δεξιού μέρους της ζυγαριάς. Έτσι έχουμε S_left = 13 και S_right = 5.
3. *Τα δύο μέρη της ζυγαριάς ισορροπούν:* Το κίβδηλο νόμισμα βρίσκεται ανάμεσα στα τελευταία x νομίσματα και από το αριστερό και από το δεξί μέρος της ζυγαριάς. Οπότε η τιμή του S_left και του S_right χρειάζονται ανανέωση. Το S_left τώρα δείχνει στο πρώτο νόμισμα από τα τελευταία x νομίσματα του αριστερού μέρους της ζυγαριάς και το S_right τώρα δείχνει στο πρώτο νόμισμα από τα τελευταία x νομίσματα του δεξιού μέρους της ζυγαριάς.
Στο παράδειγμά μας, το κίβδηλο νόμισμα βρίσκεται ανάμεσα στα τελευταία 4 νομίσματα από κάθε μέρος της ζυγαριάς (9,10,11,12,21,22,23,24). Συνεπώς ανανεώνεται η τιμή του S_left και του S_right. Έτσι έχουμε S_left = 9 και S_right = 21.

Η τιμή του x μειώνεται καθώς κατεβαίνουμε επίπεδα στη δομή του δένδρου απόφασης. Αν η τιμή του x γίνει ίση με ½ τότε μόνο δύο νομίσματα έχουν μείνει αζύγιστα. Μετά το επόμενο ζύγισμα καλούμε τη διαδικασία ελέγχου (check) με αυτά τα δύο αζύγιστα νομίσματα και ένα γνωστό κανονικό νόμισμα. Αν η τιμή του x είναι μεγαλύτερη από ½, η συνάρτηση less_than καλείται επαναληπτικά με τις ανανεωμένες τιμές S_right και S_left. Η περίπτωση 3 δε μπορεί να συμβεί παραπάνω από μία φορές.

Η Διαδικασία greater_than

Αυτή η διαδικασία είναι ανάλογη της προηγούμενης διαδικασίας. Εδώ, διατηρούμε τα πρώτα x νομίσματα στο αριστερό μέρος της ζυγαριάς (αρχίζοντας από το S_left) και ανταλλάσσουμε τα επόμενα x νομίσματα με τα πρώτα x νομίσματα του δεξιού μέρους της ζυγαριάς (αρχίζοντας από το S_right), και διατηρούμε τα επόμενα x νομίσματα του δεξιού μέρους ως έχουν. Έτσι, σε αυτή τη σύγκριση, τα τελευταία x νομίσματα και από τα δύο μέρη της ζυγαριάς δε λαμβάνονται υπόψη.
Στο παράδειγμα μας, διατηρούμε τα πρώτα 4 νομίσματα (1,2,3,4) στο αριστερό μέρος της ζυγαριάς και ανταλλάσσουμε τα επόμενα 4 νομίσματα (5,6,7,8) με τα πρώτα 4 νομίσματα που υπάρχουν στο δεξί μέρος της ζυγαριάς (13,14,15,16), και διατηρούμε τα επόμενα 4 νομίσματα του δεξιού μέρους της ζυγαριάς (17,18,19,29) όπως ήταν. Έτσι, σε αυτή τη σύγκριση, τα τελευταία 8 νομίσματα και από τα δύο μέρη της ζυγαριάς δε

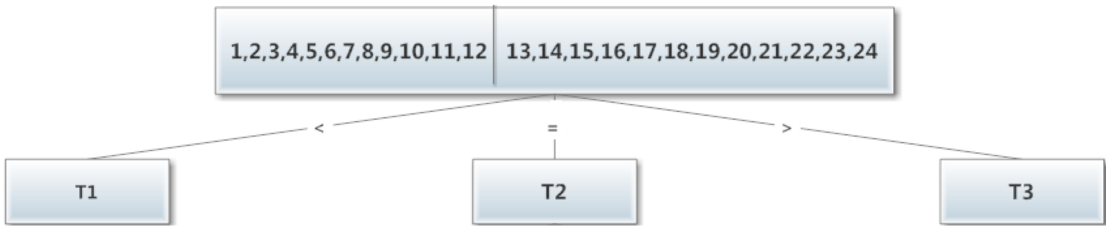
λαμβάνονται υπόψη (δηλαδή τα νομίσματα 9,10,11,12 του αριστερού μέρους της ζυγαριάς και τα νομίσματα 21,22,23,24 του δεξιού μέρους της ζυγαριάς). Και σε αυτή την περίπτωση έχουμε τρία δυνατά αποτελέσματα:

1. *Το αριστερό μέρος είναι βαρύτερο:* Το κίβδηλο νόμισμα βρίσκεται ανάμεσα στα πρώτα x νομίσματα του αριστερού μέρους, ή στα τελευταία x νομίσματα του δεξιού μέρους της ζυγαριάς. Διαιρούμε τα νομίσματα που βρίσκονται σε κάθε ένα από τα δύο μέρη της ζυγαριάς σε τέσσερα τμήματα: τα s_{l1}, s_{l2}, s_{l3} και s_{l4} για το αριστερό μέρος της ζυγαριάς, και τα s_{r1}, s_{r2}, s_{r3} και s_{r4} για το δεξί μέρος της ζυγαριάς, καθένα από τα οποία περιέχει τον ίδιο αριθμό νομισμάτων. Έπειτα, διαγράφουμε τα νομίσματα που βρίσκονται στα δύο τελευταία τμήματα του αριστερού μέρους της ζυγαριάς (δηλαδή τα s_{l3} και s_{l4}) και αυτά που βρίσκονται στα δύο πρώτα τμήματα του δεξιού μέρους της ζυγαριάς (δηλαδή τα s_{r1} και s_{r2}). Το S_left είναι τώρα το πρώτο από τα εναπομείναντα νομίσματα του αριστερού μέρους της ζυγαριάς και το S_right είναι το πρώτο από τα εναπομείναντα νομίσματα του δεξιού μέρους της ζυγαριάς. Τέλος, διατηρούμε τα νομίσματα του s_{l1} του αριστερού μέρους της ζυγαριάς και ανταλλάσσουμε τα νομίσματα του s_{l2} με τα νομίσματα του s_{r3} του δεξιού μέρους της ζυγαριάς, και διατηρούμε τα νομίσματα του s_{r4} ως έχουν.
   Στο παράδειγμά μας, το κίβδηλο νόμισμα βρίσκεται μεταξύ των 4 πρώτων νομισμάτων του αριστερού μέρους της ζυγαριάς (1,2,3,4) ή στα τελευταία 4 νομίσματα του δεξιού μέρους της ζυγαριάς (17,18,19,20). Συνεπώς διαγράφουμε τα νομίσματα (13,14,15,16) του αριστερού μέρους της ζυγαριάς και τα νομίσματα (5,6,7,8) του δεξιού μέρους της ζυγαριάς. Έτσι έχουμε S_left = 1 και S_right = 17.
2. *Το αριστερό μέρος είναι ελαφρύτερο:* Το κίβδηλο νόμισμα βρίσκεται ανάμεσα στα τελευταία x νομίσματα του αριστερού μέρους της ζυγαριάς, ή στα πρώτα x νομίσματα στο δεξί μέρος. Διαιρούμε τα νομίσματα που βρίσκονται σε κάθε ένα από τα δύο μέρη της ζυγαριάς σε τέσσερα τμήματα: τα s_{l1}, s_{l2}, s_{l3} και s_{l4} για το αριστερό μέρος της ζυγαριάς, και τα s_{r1}, s_{r2}, s_{r3} και s_{r4} για το δεξί μέρος της ζυγαριάς, καθένα από τα οποία περιέχει τον ίδιο αριθμό νομισμάτων. Έπειτα διαγράφουμε τα νομίσματα που βρίσκονται στα δύο πρώτα τμήματα του αριστερού μέρους της ζυγαριάς (δηλαδή τα s_{l1} και s_{l2}) και αυτά που βρίσκονται στα δύο τελευταία τμήματα του δεξιού μέρους της ζυγαριάς (δηλαδή τα s_{r3} και s_{r4}). Το S_left είναι τώρα το πρώτο από τα εναπομείναντα νομίσματα του αριστερού μέρους της ζυγαριάς και το S_right είναι το πρώτο από τα εναπομείναντα νομίσματα του δεξιού μέρους της ζυγαριάς. Τέλος, ανταλλάσσουμε τα νομίσματα του s_{l3} του αριστερού μέρους της ζυγαριάς με τα νομίσματα του s_{r2} του δεξιού μέρους της ζυγαριάς, και διατηρούμε τα νομίσματα του s_{l4} και του s_{r1} ως έχουν.
   Στο παράδειγμά μας, το κίβδηλο νόμισμα βρίσκεται μεταξύ των τελευταίων 4 νομισμάτων του αριστερού μέρους της ζυγαριάς (13,14,15,16) ή στα πρώτα 4 νομίσματα του δεξιού μέρους της ζυγαριάς (5,6,7,8). Συνεπώς διαγράφουμε τα νομίσματα (1,2,3,4) του αριστερού μέρους της ζυγαριάς και τα νομίσματα (17,18,19,20) του δεξιού μέρους της ζυγαριάς. Έτσι έχουμε S_left =13 και S_right = 5.
3. *Τα δύο μέρη της ζυγαριάς ισορροπούν:* Το κίβδηλο νόμισμα βρίσκεται ανάμεσα στα τελευταία x νομίσματα και του αριστερού και του δεξιού μέρους της ζυγαριάς. Οπότε οι τιμές και του S_left και του S_right χρειάζονται ανανέωση. Το S_left τώρα δείχνει στο πρώτο νόμισμα των τελευταίων x

νομισμάτων του αριστερού μέρους της ζυγαριάς και το S_right δείχνει στο πρώτο νόμισμα από τα τελευταία x νομίσματα του δεξιού μέρους της ζυγαριάς.

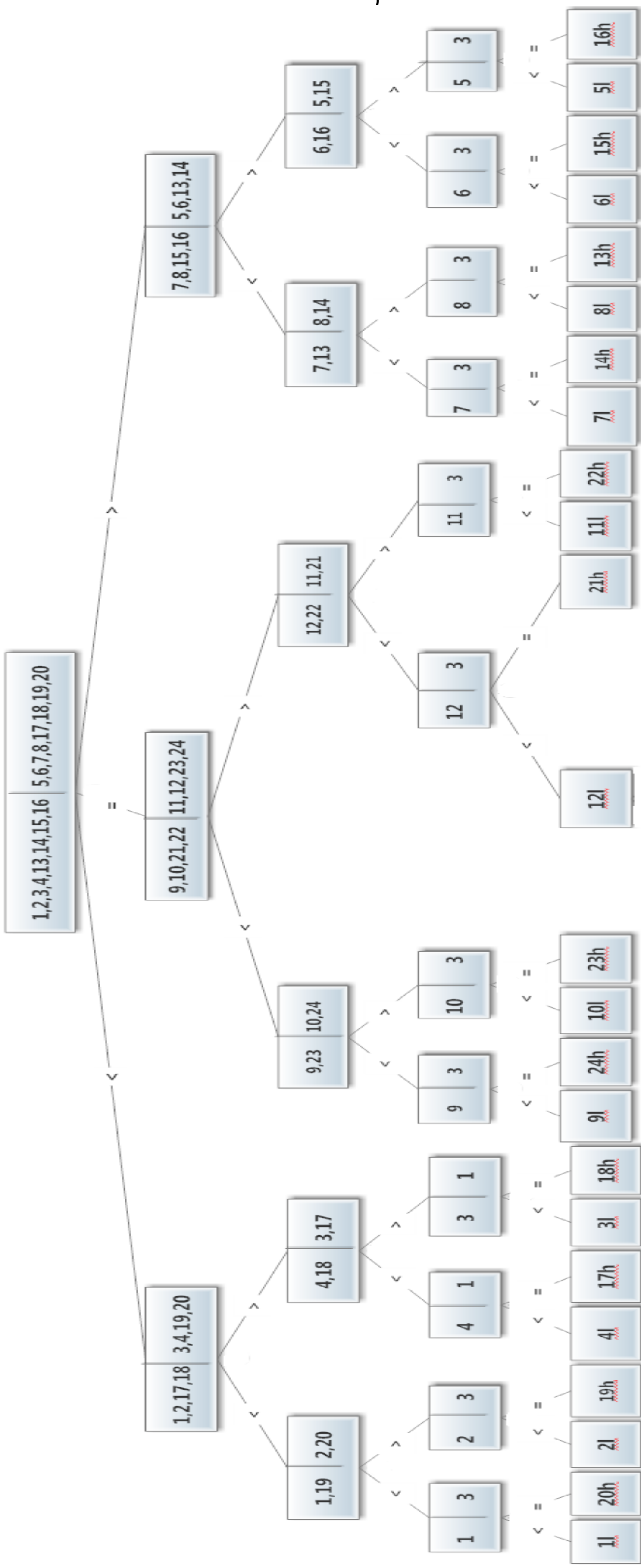
Στο παράδειγμά μας, το κίβδηλο νόμισμα βρίσκεται ανάμεσα στα τελευταία 4 νομίσματα από κάθε μέρος της ζυγαριάς (9,10,11,12,21,22,23,24). Επομένως, έχουμε ανανέωση στην τιμή του S_left και του S_right. Τώρα έχουμε S_left = 9 και S_right = 21.

Η τιμή του x μειώνεται καθώς κατεβαίνουμε επίπεδα στη δομή του δένδρου απόφασης. Αν η τιμή του x γίνει ίση με ½ μόνο δύο νομίσματα θα μείνουν απροσδιόριστα. Σε αυτό το σημείο καλούμε τη διαδικασία ελέγχου (check) με αυτά τα δύο αζύγιστα νομίσματα κι ένα γνωστό κανονικό νόμισμα. Αυτή η διαδικασία ελέγχου βρίσκει το κίβδηλο νόμισμα από τη σύγκριση αυτών των τριών νομισμάτων. Αν η τιμή του x είναι μεγαλύτερη από ½, τότε η διαδικασία greater_than καλείται επαναληπτικά με τις ανανεωμένες τιμές των S_left και S_right. Και στην περίπτωση του greater_than η περίπτωση 3 δε μπορεί να συμβεί παραπάνω από μία φορά.

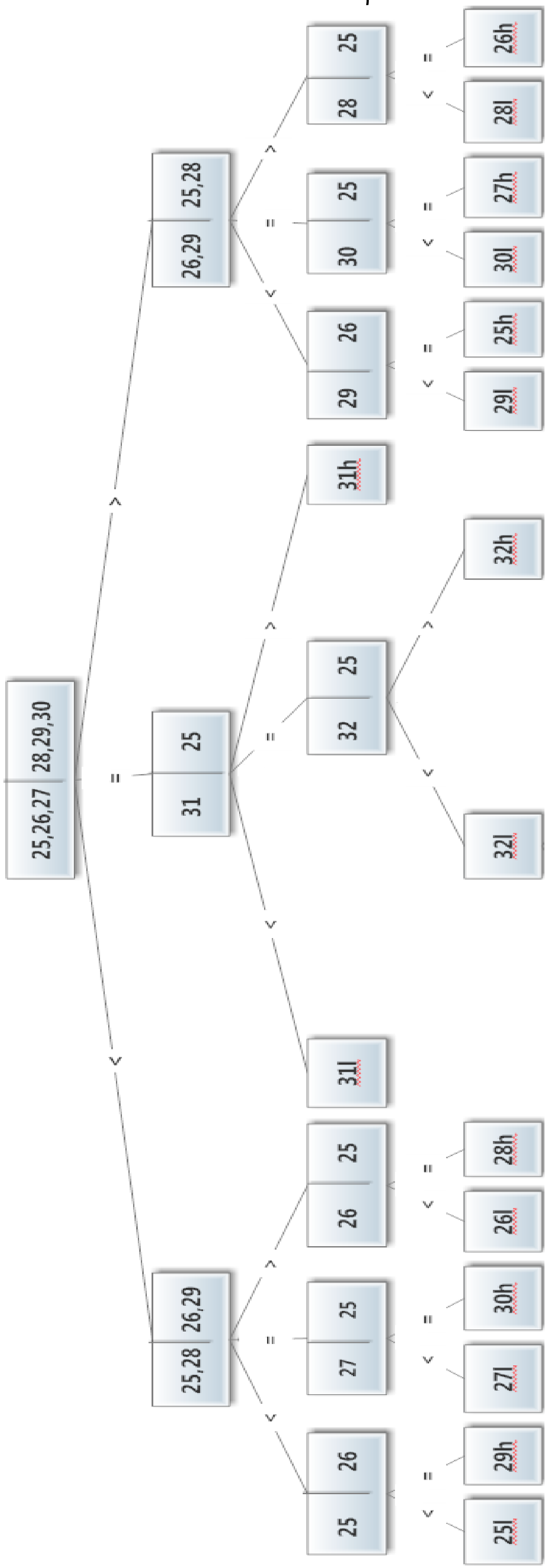


όπου T1, T2, T3 υποδέντρα.

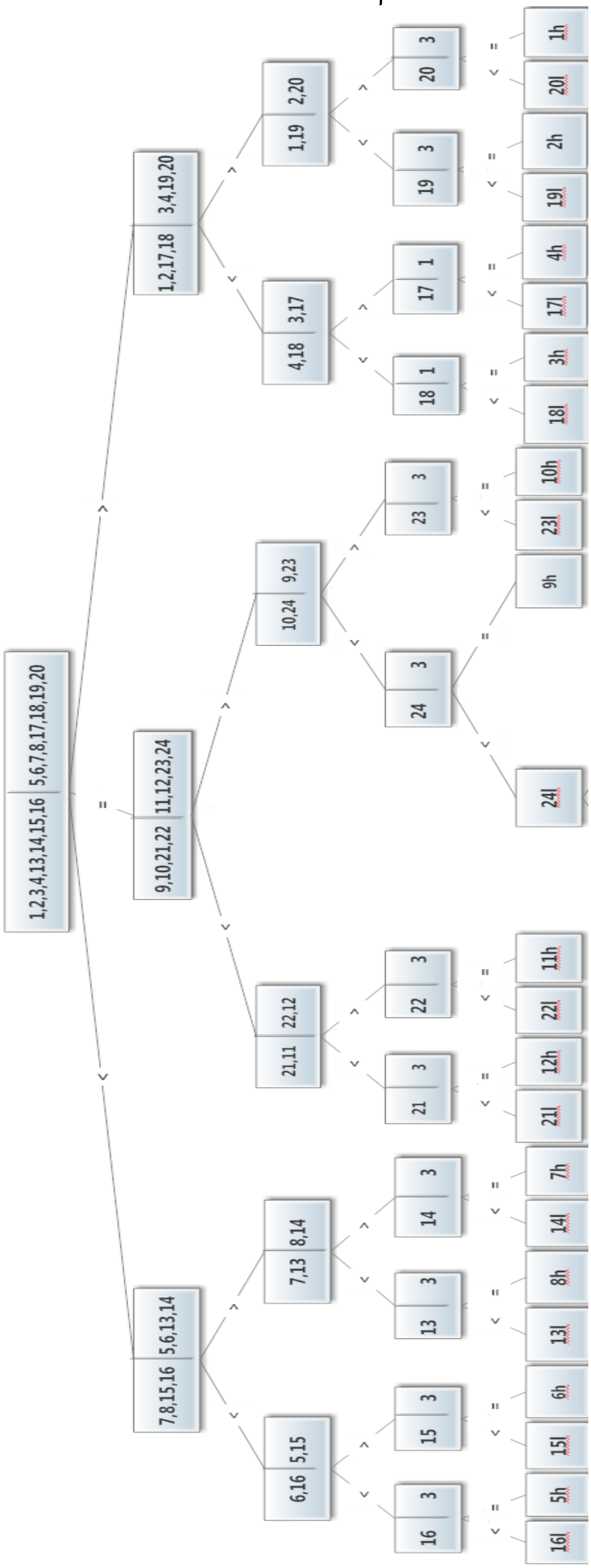
Το Τ1 είναι το ακόλουθο υποδέντρο:



Το T2 είναι το ακόλουθο υποδέντρο:



Το Τ3 είναι το ακόλουθο υποδέντρο:



3.3.2 Ένας αλγόριθμος για την επίλυση του προβλήματος των n νομισμάτων όπου n άρτιος, $n \geq 6$

Σε αυτή την εκδοχή του αλγορίθμου, πρώτα βρίσκουμε τη μικρότερη δυνατή τιμή του y βασιζόμενη στον αριθμό n που μας δίνεται, έτσι ώστε $y \geq n$, όπου το y είναι μια δύναμη του 2. Αν ισχύει $y = n$, αυτή η εκδοχή του αλγορίθμου ταιριάζει απόλυτα με τα βήματα που περιγράφηκαν στην ενότητα Α.

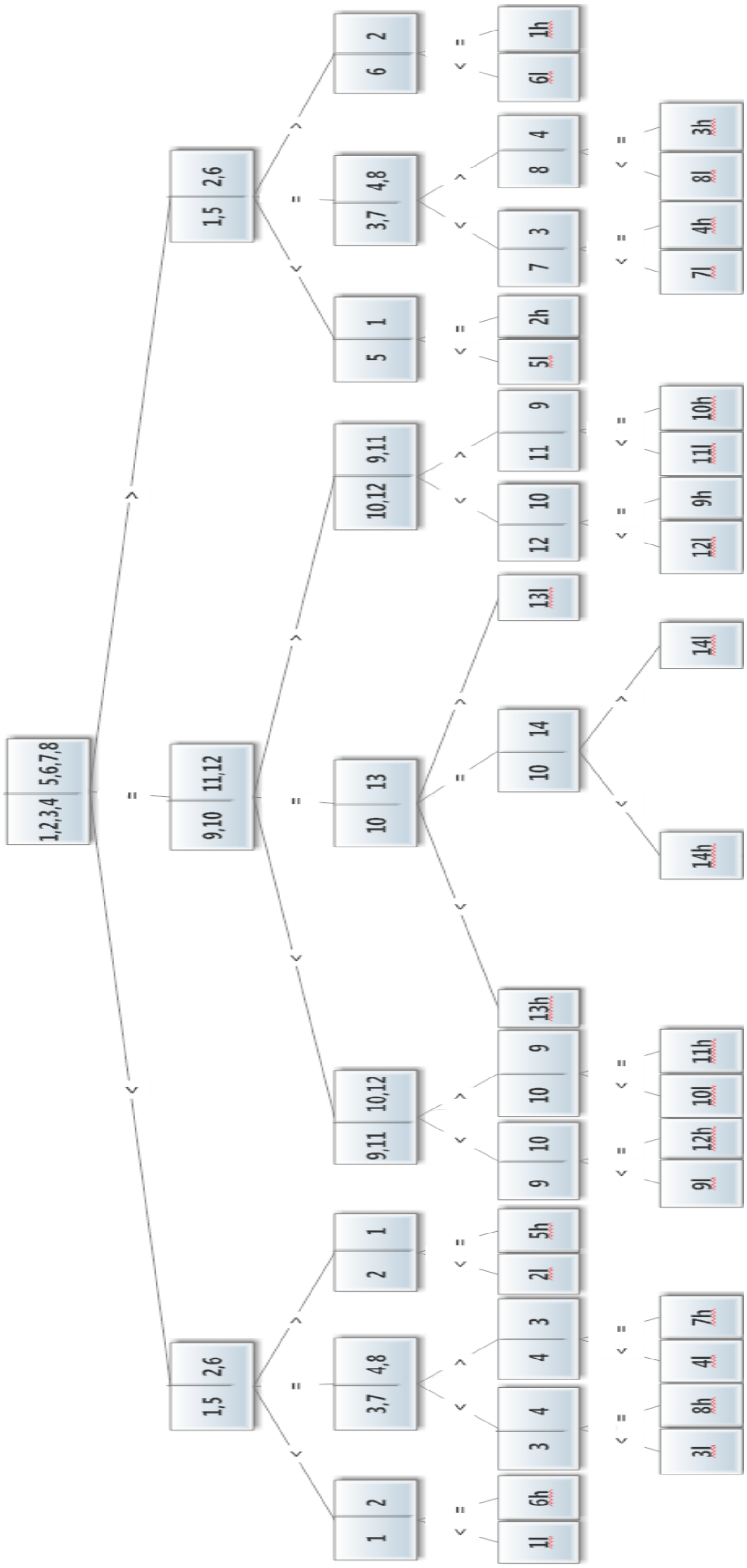
Από την άλλη, αν το n δεν είναι ίσο με το y, οπότε $n < y$ (με $2^{p-1} < n < 2^p=y$, για κάποια ακέραια τιμή του $p \geq 3$), τότε ο αριθμός των νομισμάτων που συγκρίνονται στο κάθε μέρος της ζυγαριάς ισούται με 2x, όπου $x = y/8$. Για παράδειγμα, αν είναι $n = 10$, τότε $y = 16$ και $x = 16/8 = 2$, και ο συνολικός αριθμός νομισμάτων που συγκρίνονται στη ρίζα του δένδρου απόφασης είναι 8 θέτοντας τα πρώτα 4 νομίσματα στο αριστερό μέρος της ζυγαριάς και τα επόμενα 4 στο δεξί, αν $n = 22$, τότε έχουμε $y = 32$ και $x = 4$, και ο συνολικός αριθμός νομισμάτων που συγκρίνονται στη ρίζα του δένδρου απόφασης είναι 16, θέτοντας τα 8 πρώτα νομίσματα στο αριστερό μέρος της ζυγαριάς και τα επόμενα 8 στο δεξί, κ.ο.κ.
Στο παράδειγμα που ακολουθεί έχουμε $n = 14$, οπότε $y = 16 > n$ και $x = 16/8 = 2$. Έτσι ο συνολικός αριθμός νομισμάτων που συγκρίνεται στη ρίζα του δένδρου απόφασης είναι 8, θέτοντας τα πρώτα 4 νομίσματα στο αριστερό μέρος της ζυγαριάς (1,2,3,4) και τα επόμενα 4 νομίσματα στο δεξί μέρος της ζυγαριάς (5,6,7,8).

Αν στην πρώτη σύγκριση η ζυγαριά ισορροπεί, τότε το κίβδηλο νόμισμα βρίσκεται ανάμεσα στα εναπομείναντα $n – 4x$ νομίσματα. Για παράδειγμα, αν έχουμε ισότητα για $n = 10$, τότε το κίβδηλο νόμισμα ανήκει στα εναπομείναντα 2 νομίσματα, για $n = 22$, το κίβδηλο νόμισμα ανήκει στα εναπομείναντα 6 νομίσματα, κ.ο.κ.
Στο παράδειγμά μας, αν στην πρώτη σύγκριση η ζυγαριά ισορροπεί, τότε το κίβδηλο βρίσκεται στα εναπομείναντα 6 νομίσματα (9,10,11,12, 13, 14).

Από την άλλη πλευρά, αν το αριστερό μέρος της ζυγαριάς είναι ελαφρύτερο από το δεξί μέρος, τότε το κίβδηλο νόμισμα βρίσκεται ανάμεσα στα πρώτα 4x νομίσματα, από τα οποία ένα από τα πρώτα 2x νομίσματα μπορεί να είναι ελαφρύτερο ή ένα από τα επόμενα 2x νομίσματα μπορεί να είναι βαρύτερο. Τότε καλείται η διαδικασία less_than για το αριστερό υποδέντρο του δένδρου απόφασης.
Στο παράδειγμά μας, αν το αριστερό μέρος της ζυγαριάς είναι ελαφρύτερο από το δεξί, τότε το κίβδηλο βρίσκεται ανάμεσα στα 8 πρώτα νομίσματα, από τα οποία ένα από τα πρώτα 4 νομίσματα (1,2,3,4) μπορεί να είναι ελαφρύτερο ή ένα από τα επόμενα 4 νομίσματα (5,6,7,8) μπορεί να είναι βαρύτερο.

Αν το αριστερό μέρος της ζυγαριάς είναι βαρύτερο από το δεξί, τότε το κίβδηλο νόμισμα βρίσκεται ανάμεσα στα πρώτα 4x νομίσματα, από τα οποία ένα από τα πρώτα 2x νομίσματα μπορεί να είναι βαρύτερο ή ένα από τα επόμενα 2x νομίσματα μπορεί να είναι ελαφρύτερο. Τότε καλείται η διαδικασία greater_than για το δεξί υποδέντρο του δένδρου απόφασης.
Στο παράδειγμά μας, αν το αριστερό μέρος της ζυγαριάς είναι βαρύτερο από το δεξί, τότε το κίβδηλο νόμισμα βρίσκεται ανάμεσα στα πρώτα 8 νομίσματα, από τα οποία ένα από τα πρώτα 4 νομίσματα (1,2,3,4) μπορεί να είναι βαρύτερο ή ένα από τα επόμενα 4 νομίσματα (5,6,7,8) μπορεί να είναι ελαφρύτερο.

Με αυτό τον τρόπο, κάθε φορά που κινούμαστε προς τα φύλλα δένδρου απόφασης η τιμή του x διαιρείται με το 2. Όταν έχουμε x = ½, υπάρχουν μόνο δύο νομίσματα που δεν έχουν ζυγιστεί, ένα εκ των οποίων είναι το κίβδηλο νόμισμα. Στο σημείο αυτό η διαδικασία ελέγχου (check) βρίσκει το κίβδηλο νόμισμα, και επίσης προσδιορίζει αν είναι βαρύτερο ή ελαφρύτερο από τα υπόλοιπα.

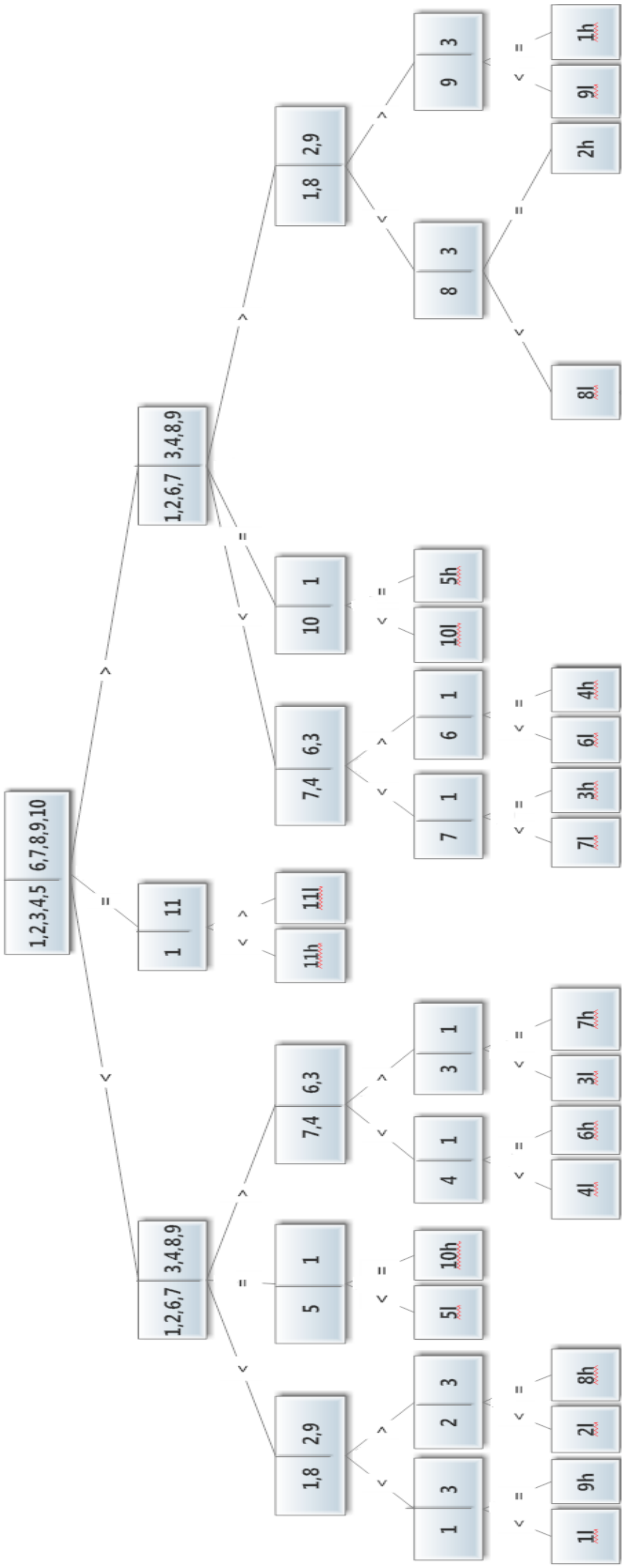


3.3.3 Ένας αλγόριθμος για την επίλυση του προβλήματος των n νομισμάτων όπου n περιττός, n ≥ 7

Αν το n είναι περιττός αριθμός, τότε μπορούμε να γράψουμε n = 2m + 1, όπου το 2m προφανώς είναι ένας άρτιος αριθμός. Έπειτα διαιρούμε αυτά τα 2m νομίσματα σε δύο ομάδες, που έχει η καθεμιά m νομίσματα. Καθεμία από τις ομάδες των m νομισμάτων τίθεται σε κάθε ένα από τα δύο μέρη της ζυγαριάς. Αν η ζυγαριά ισορροπεί, τότε το κίβδηλο νόμισμα θα είναι το τελευταίο νόμισμα. Αλλά ακόμη δεν γνωρίζουμε αν αυτό είναι βαρύτερο ή ελαφρύτερο από τα υπόλοιπα νομίσματα. Γι' αυτό συγκρίνουμε το τελευταίο νόμισμα με το πρώτο, το οποίο γνωρίζουμε ότι είναι κανονικό. Αν το τελευταίο νόμισμα είναι ελαφρύτερο από το κανονικό νόμισμα, συμπεραίνουμε ότι είναι ελαφρύτερο από τα υπόλοιπα νομίσματα, αλλιώς είναι βαρύτερο.

Από την άλλη μεριά, η ζυγαριά μπορεί να δείξει ανισότητα, όπου σε κάθε μεριά της ζυγαριάς έχουμε m νομίσματα, και το κίβδηλο νόμισμα περιέχεται ανάμεσα στα πρώτα n – 1 = 2m νομίσματα, που είναι η περίπτωση εύρεσης κίβδηλου νομίσματος για μια δοθείσα ομάδα άρτιου αριθμού νομισμάτων. Αυτός ο αλγόριθμος έχει αναπτυχθεί σε προηγούμενη ενότητα.

*Στο παράδειγμα που ακολουθεί, είναι n = 11, οπότε έχουμε n = 2*5+1. Συνεπώς θέτουμε m = 5 νομίσματα σε κάθε ένα από τα δύο μέρη της ζυγαριάς, 1,2,…,5 στο αριστερό μέρος της ζυγαριάς και 6,7,…,10 στο δεξί μέρος της ζυγαριάς. Στην περίπτωση που η ζυγαριά ισορροπήσει, το κίβδηλο νόμισμα θα είναι το τελευταίο (11). Αλλά ακόμη δεν γνωρίζουμε αν αυτό είναι βαρύτερο ή ελαφρύτερο από τα υπόλοιπα. Έτσι συγκρίνουμε το 11 με ένα νόμισμα που γνωρίζουμε ότι είναι κανονικό και βρίσκουμε αν είναι ελαφρύτερο ή βαρύτερο από τα υπόλοιπα. Στην περίπτωση που η ζυγαριά δείξει ανισότητα τότε συμπεραίνουμε ότι το κίβδηλο νόμισμα περιέχεται ανάμεσα στα 10 πρώτα νομίσματα, που είναι η περίπτωση εύρεσης κίβδηλου νομίσματος για μια δοθείσα ομάδα άρτιου αριθμού νομισμάτων, που δεν είναι δύναμη του 2, (περίπτωση Β).*



1,2,3,4,5 | 6,7,8,9,10
1,2,6,7 | 3,4,8,9
1 | 11
1,2,6,7 | 3,4,8,9
1,8 | 2,9
5 | 1
7,4 | 6,3
11h
11l
7,4 | 6,3
10 | 1
1,8 | 2,9
1 | 3
2 | 3
5l
10h
4 | 1
3 | 1
7 | 1
6 | 1
10l
5h
8 | 3
9 | 3
1l
9h
2l
8h
4l
6h
3l
7h
7l
3h
6l
4h
8l
2h
9l
1h

Κεφάλαιο 4

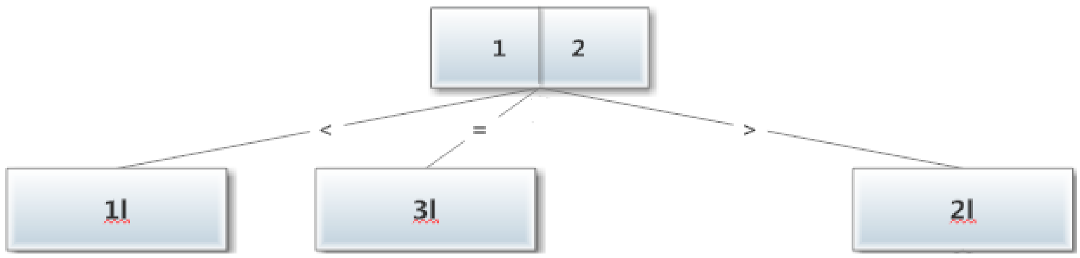
Κάποιες Άλλες Εκδοχές του Προβλήματος των Κίβδηλων Νομισμάτων

Παρακάτω, δίνουμε διάφορες άλλες εκδοχές του προβλήματος ενός κίβδηλου νομίσματος, με τις αντίστοιχες απαντήσεις σε συγκεκριμένες περιπτώσεις.

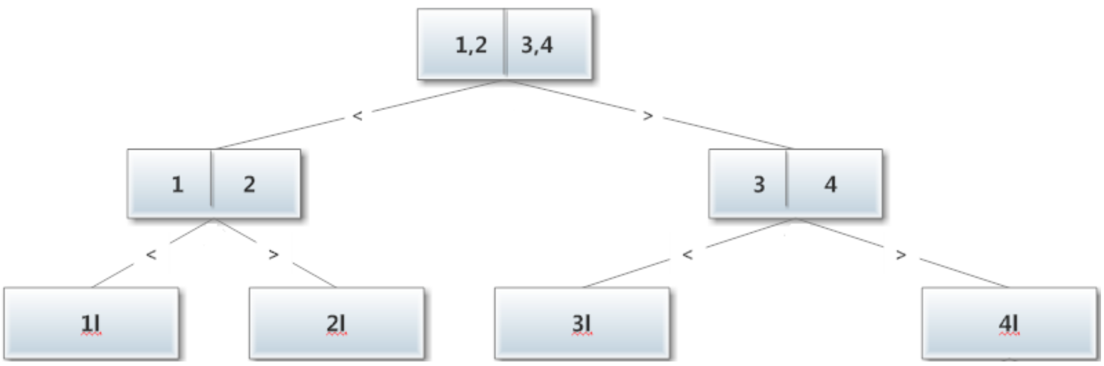
4.1 Δίδονται n νομίσματα ένα εκ των οποίων είναι κίβδηλο και ελαφρύτερο από τα άλλα.

Παραθέτουμε την απάντηση στο πρόβλημα για n = 3,4,…,12.

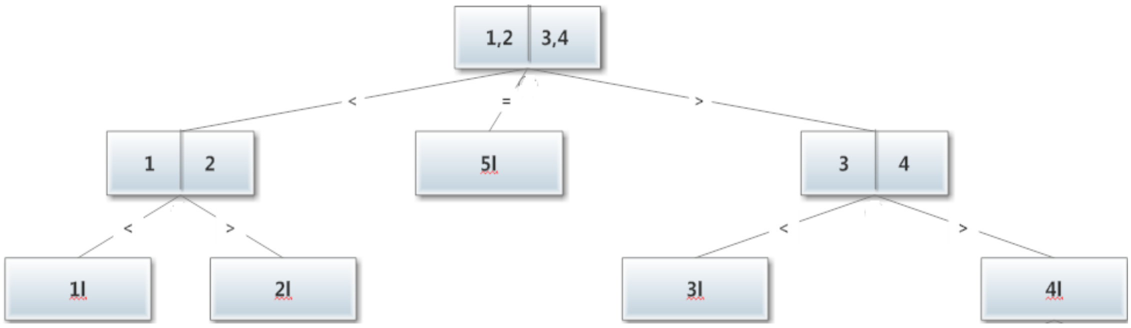
- Για n = 3.



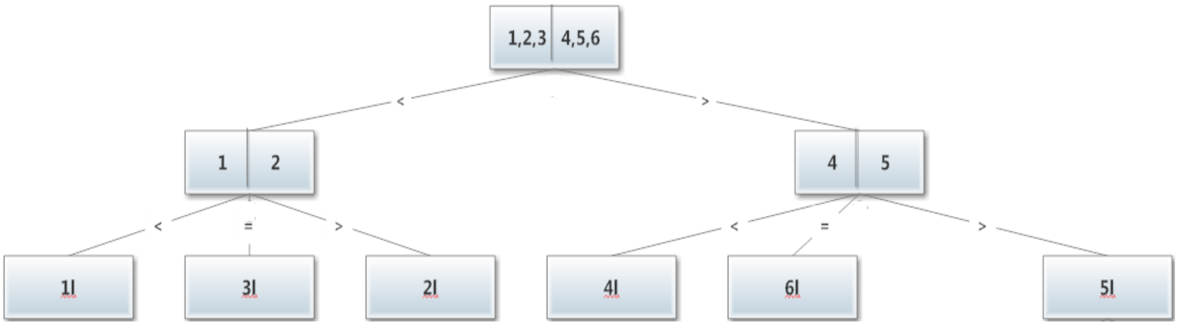
- Για n = 4.



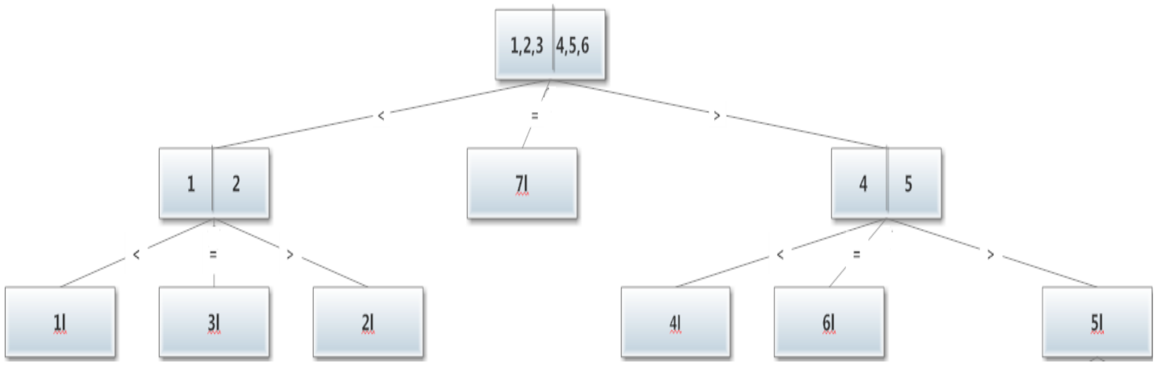
- Για n = 5.



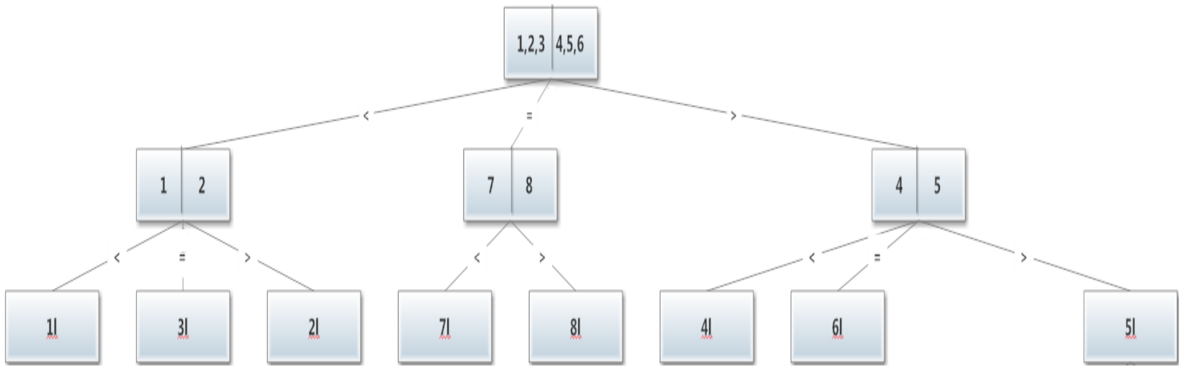
- Για n = 6.



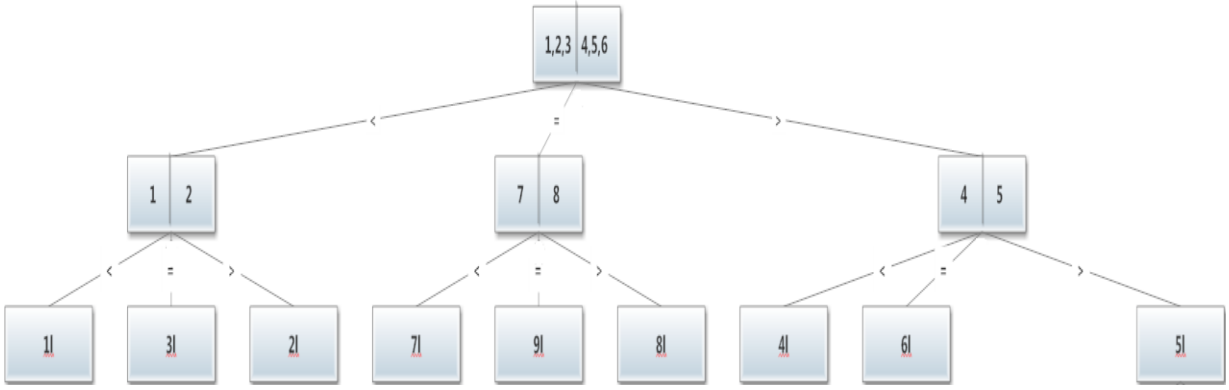
- Για n = 7.



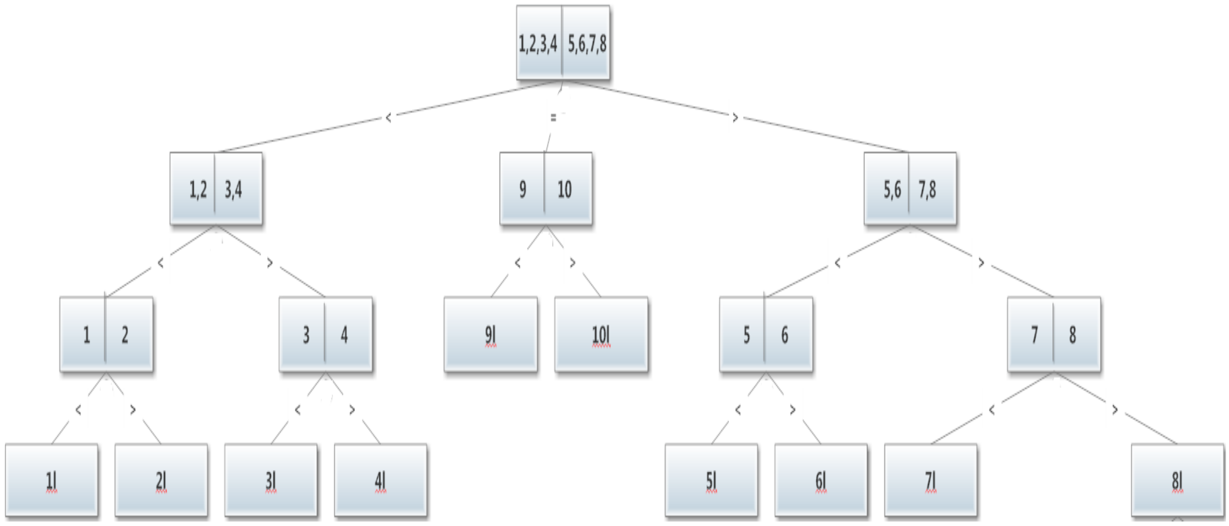
- Για n = 8.


1,2,3
4,5,6
<
=
>
1
2
7
8
4
5
<
=
>
<
>
<
=
>
1l
3l
2l
7l
8l
4l
6l
5l

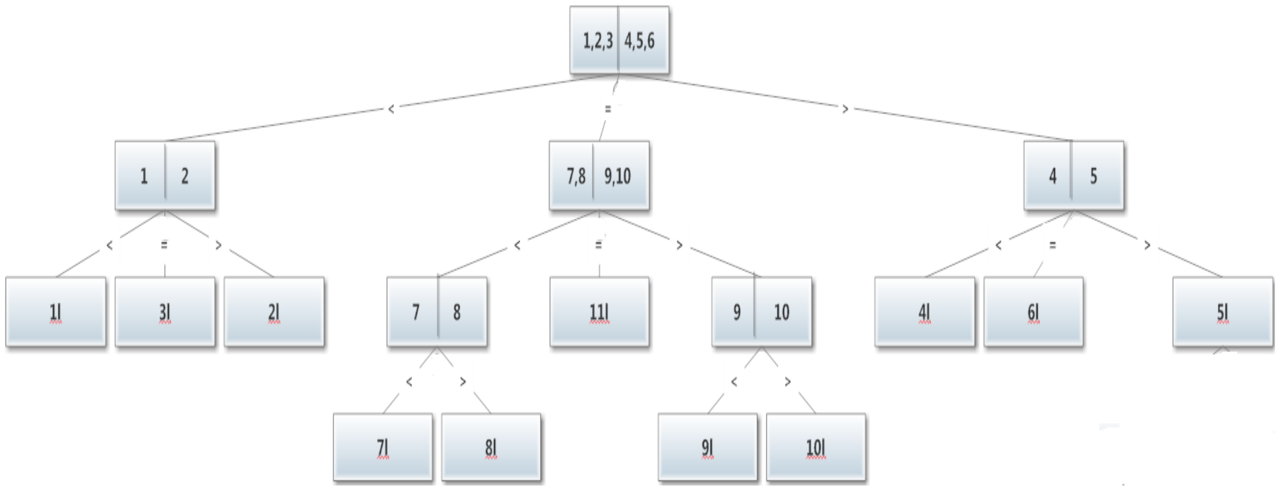
- Για n = 9.


1,2,3
4,5,6
<
=
>
1
2
7
8
4
5
<
=
>
<
=
>
<
=
>
1l
3l
2l
7l
9l
8l
4l
6l
5l

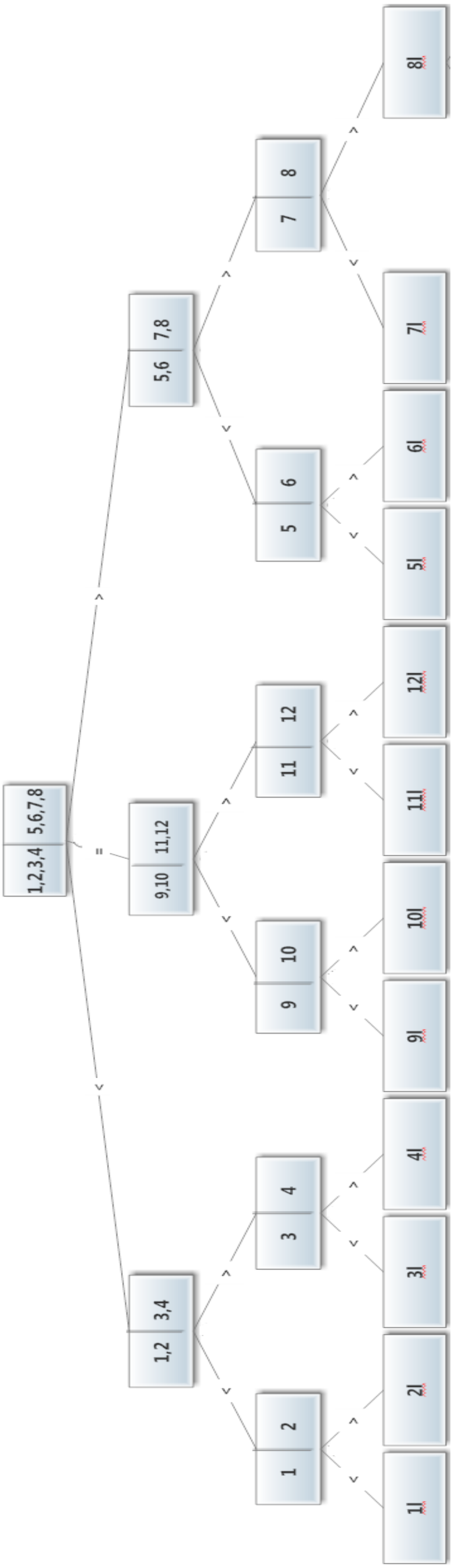
- Για n = 10.



- Για n = 11.



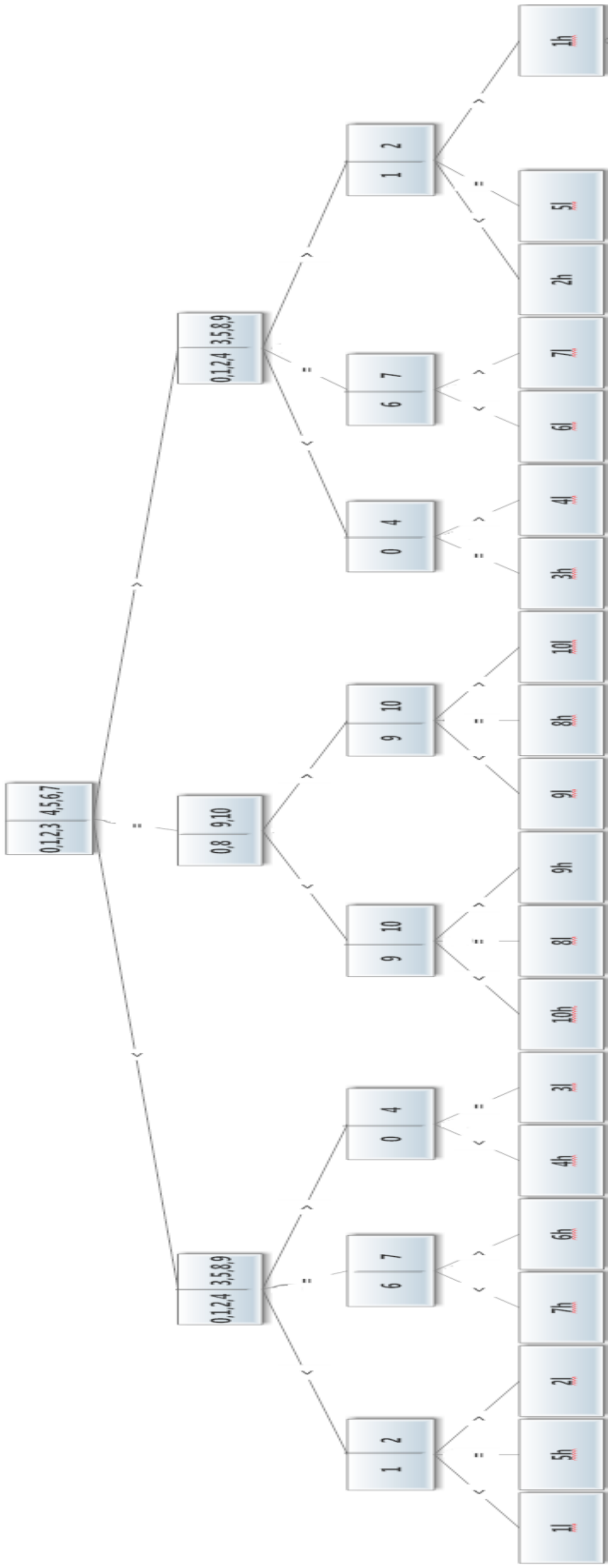
- Για n = 12.



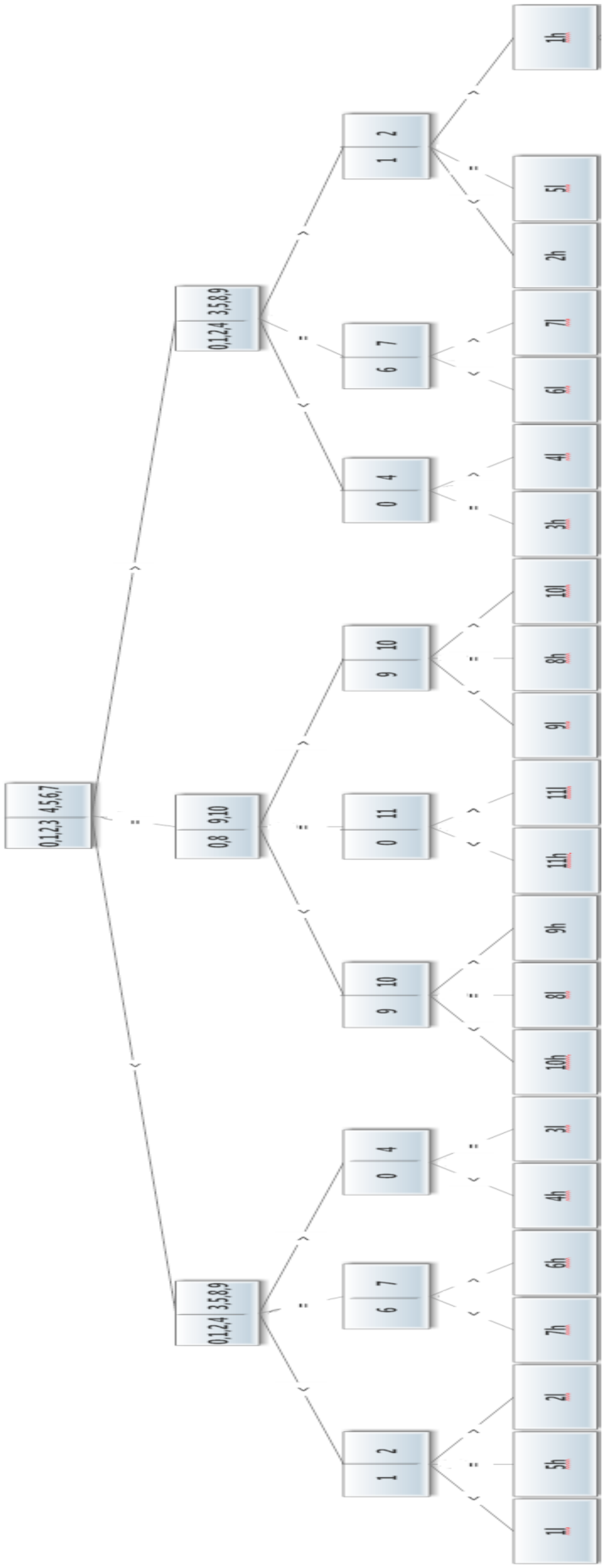
4.2 Δίδονται n νομίσματα (0,1,2,…,n-1), όπου το 0 είναι κανονικό κι ένα από τα υπόλοιπα είναι κίβδηλο αλλά δεν είναι γνωστό αν είναι βαρύτερο ή ελαφρύτερο.

Παραθέτουμε την απάντηση στο πρόβλημα για n = 11,12,13 και 14.

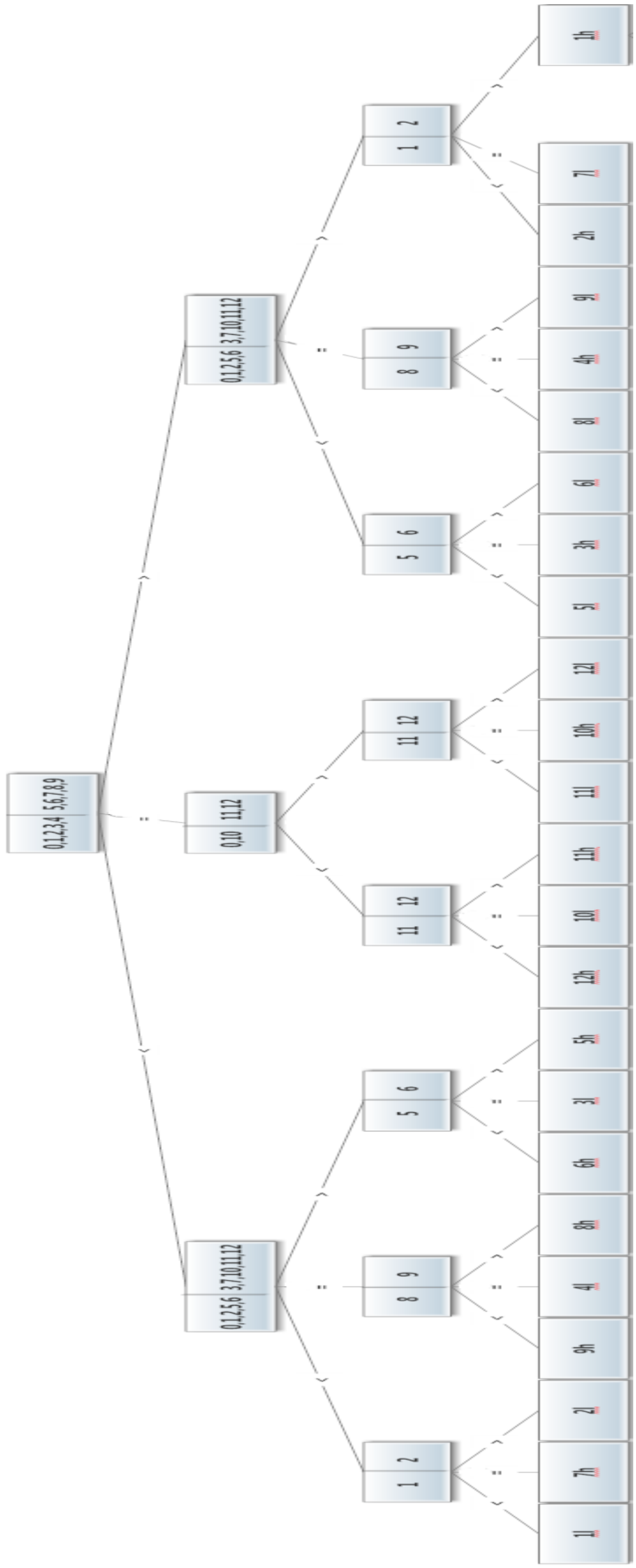
- Για n = 11.



- Για n=12.

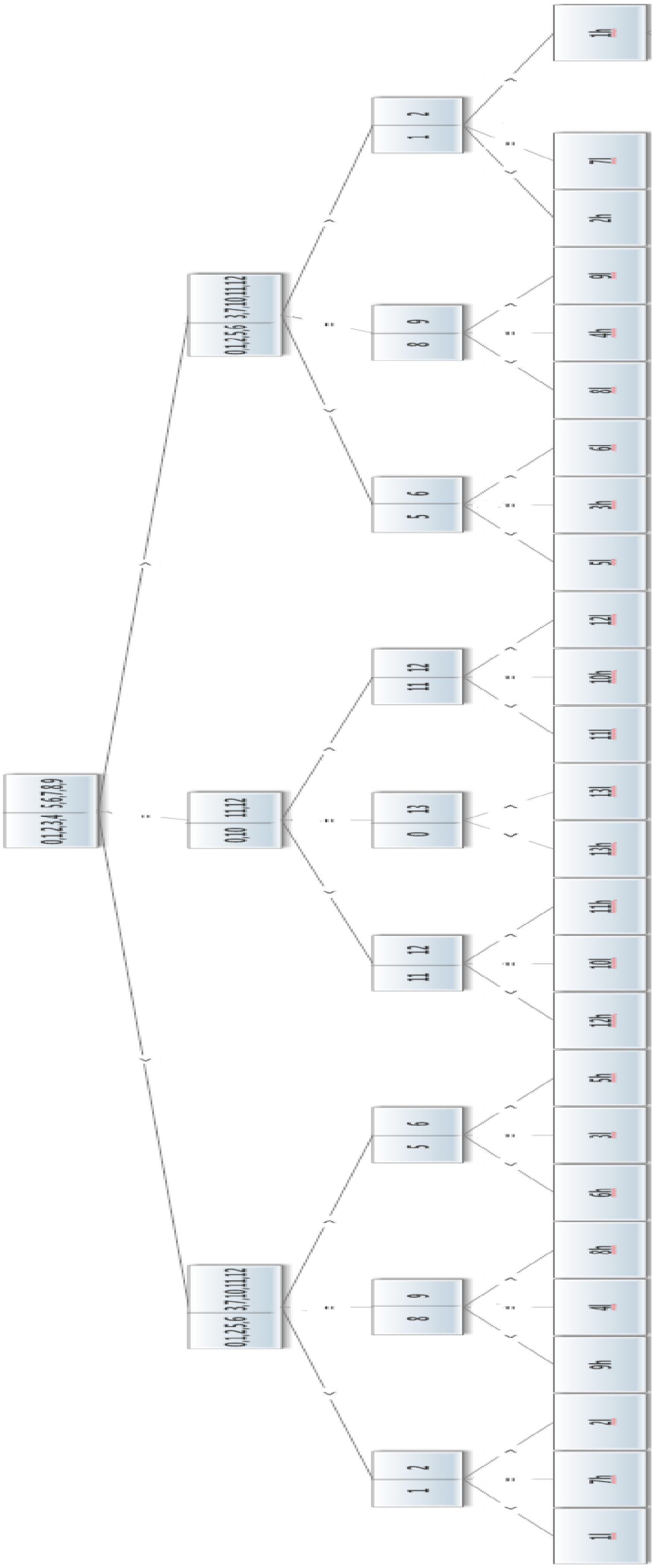


- Για n=13.



0,1,2,3,4 5,6,7,8,9
0,1,2,5,6 3,7,10,11,12
0,10 11,12
0,1,2,5,6 3,7,10,11,12
1 2
8 9
5 6
11 12
11 12
5 6
8 9
1 2
1h
7l
2h
9l
4h
8l
6l
3h
5l
12l
10h
11l
11h
10l
12h
5h
3l
6h
8h
4l
9h
2l
7h
1l

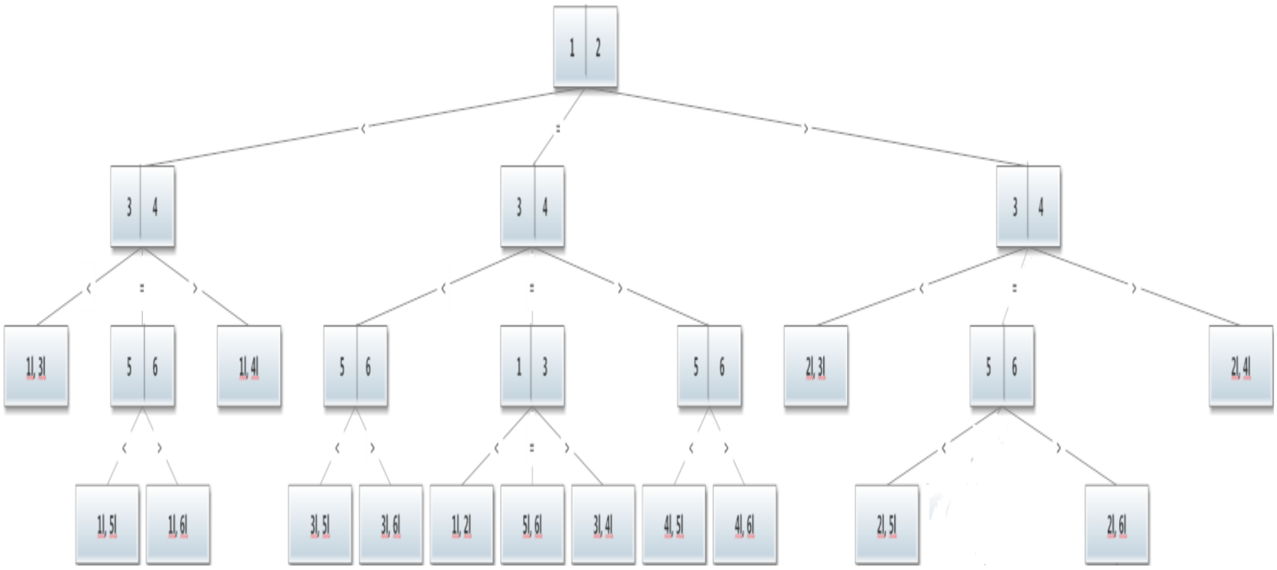
- Για n = 14.



4.3 Το πρόβλημα των δύο κίβδηλων νομισμάτων

Δεδομένος συνολικός αριθμός νομισμάτων n, εκ των οποίων τα δύο είναι ελαφρύτερα από τα υπόλοιπα και του ίδιου βάρους μεταξύ τους.

- Για n = 6.



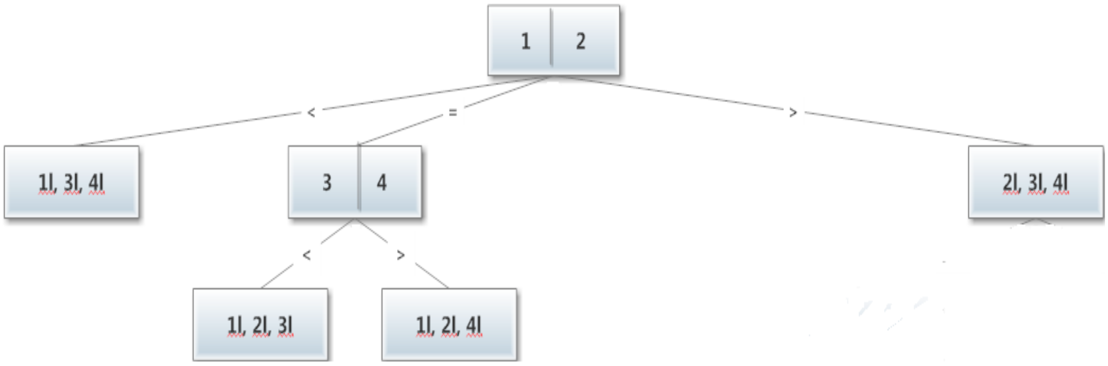
Παρατήρηση: Η υπόθεση ότι τα δύο (ελαφρύτερα) κίβδηλα έχουν ίδιο βάρος μεταξύ τους, χρειάζεται στην περίπτωση 1 < 2, 3 = 4 (και αντίστοιχα στην 1 > 2 και 3 = 4) διότι διαφορετικά η περίπτωση 5 = 6 δεν θα μπορούσε να αποκλειστεί: Θα έδινε 1l, 2l.

4.4 Το πρόβλημα των τριών κίβδηλων νομισμάτων

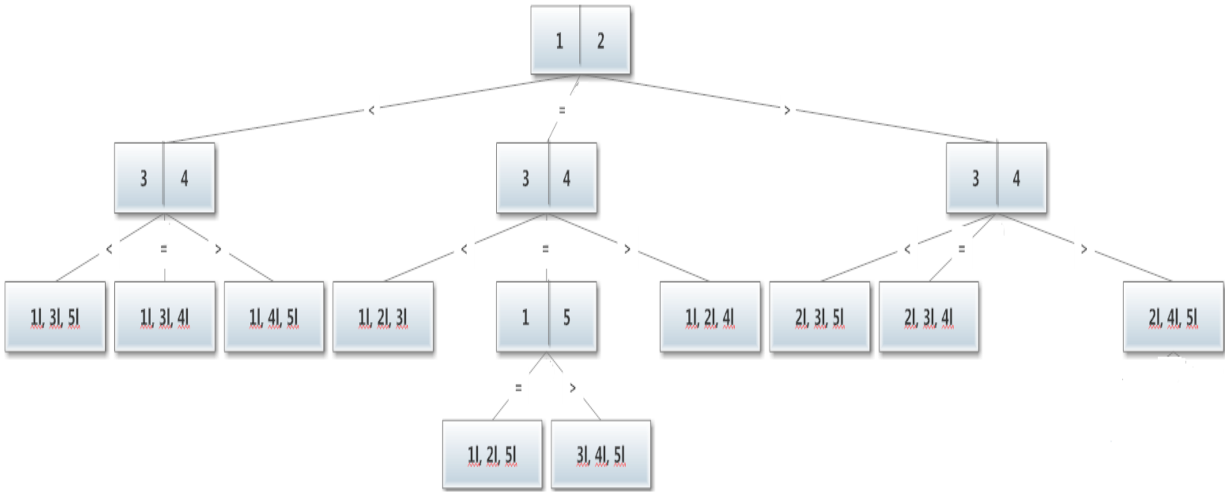
Δεδομένος συνολικός αριθμός νομισμάτων n, εκ των οποίων τα τρία είναι ελαφρύτερα από τα υπόλοιπα και του ίδιου βάρους μεταξύ τους.

Παραθέτουμε την απάντηση στο πρόβλημα για n = 4, 5 και 6.

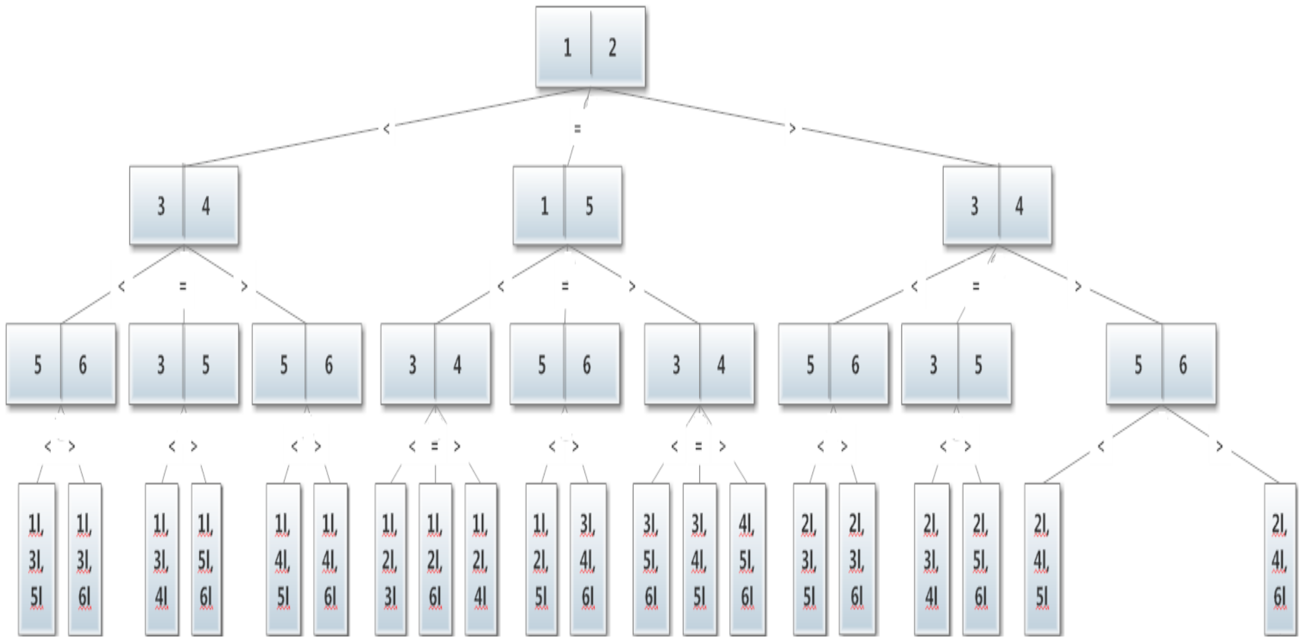
- Για n = 4.


1 | 2
<
=
>
1l, 3l, 4l
3 | 4
2l, 3l, 4l
<
>
1l, 2l, 3l
1l, 2l, 4l

- Για n = 5.


1 | 2
<
=
>
3 | 4
3 | 4
3 | 4
<
=
>
<
=
>
<
=
>
1l, 3l, 5l
1l, 3l, 4l
1l, 4l, 5l
1l, 2l, 3l
1 | 5
1l, 2l, 4l
2l, 3l, 5l
2l, 3l, 4l
2l, 4l, 5l
=
>
1l, 2l, 5l
3l, 4l, 5l

• Για n = 6.



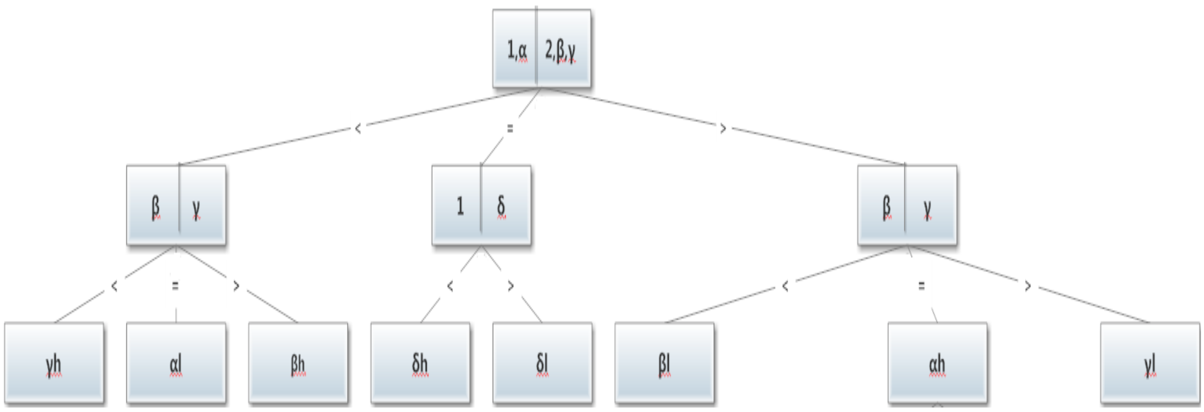
Παρατήρηση: Προφανώς, και πάλι η υπόθεση ότι τα τρία (ελαφρύτερα) κίβδηλα έχουν το ίδιο βάρος μεταξύ τους είναι απαραίτητη για τις παραπάνω λύσεις. Για παράδειγμα, στην περίπτωση n = 4, το 1 < 2 δεν θα αρκούσε για να συμπεράνουμε 1l, 3l, 4l, αφού θα μπορούσε να είναι κίβδηλα τα 1, 2, 3 και κανονικό το 4 (1 < 2 < 3 < 4).

Κεφάλαιο 5

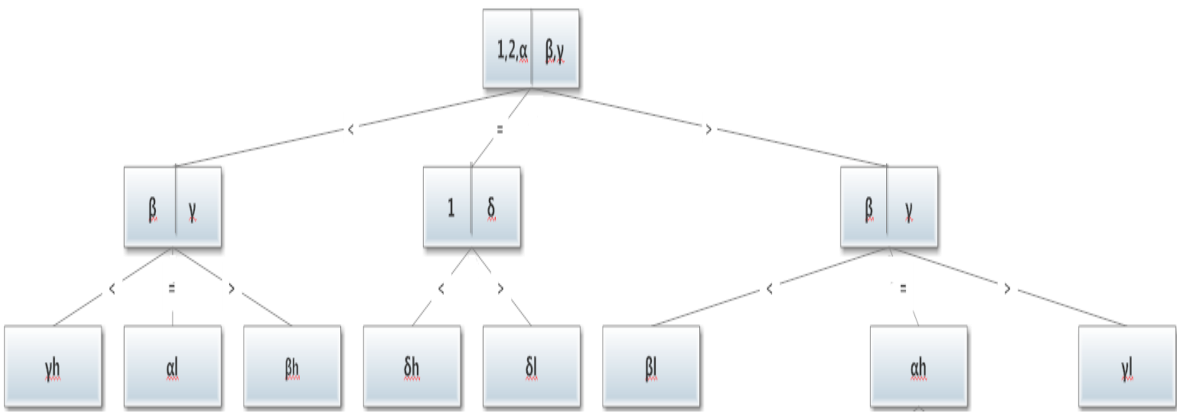
Επίλυση του Προβλήματος των Κίβδηλων Νομισμάτων με τη Δημιουργία Ομάδων

Έχουμε n νομίσματα εκ των οποίων γνωρίζουμε ότι δύο (τα 1,2) είναι κανονικά και ότι ένα είναι κίβδηλο και δεν γνωρίζουμε αν είναι ελαφρύτερο ή βαρύτερο από τα υπόλοιπα νομίσματα. Θέλουμε να δημιουργήσουμε ομάδες που να έχουν μία από τις δύο ακόλουθες μορφές:

Α.



Β.



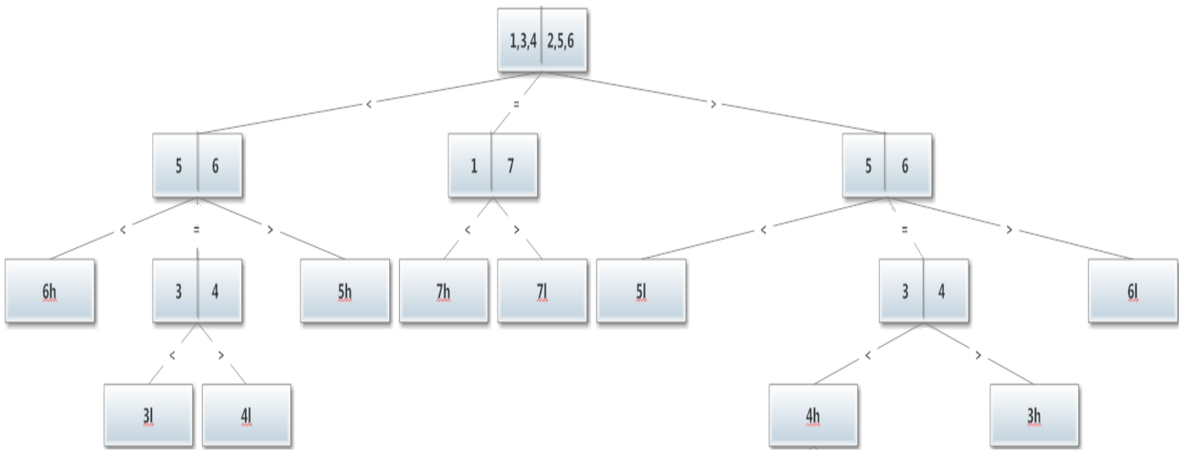
Σε κάθε μέρος της ζυγαριάς πρέπει να έχουμε τον ίδιο αριθμό νομισμάτων. Ο αριθμός νομισμάτων που θα έχει κάθε ομάδα (α, β, γ, δ) καθώς και το ποια από τις δύο παραπάνω μορφές θα επιλέξουμε εξαρτάται από το πλήθος των νομισμάτων n. Μπορούμε να διακρίνουμε τις ακόλουθες τέσσερις περιπτώσεις:

i. Όταν είναι το n είναι πολλαπλάσιο του 4, n = 4k, θα επιλέξουμε την μορφή Α, όπου στο πρώτο ζύγισμα σε κάθε μέρος της ζυγαριάς έχουμε ένα κανονικό νόμισμα (το 0 βρίσκεται στο αριστερό μέρος της ζυγαριάς ενώ το 1 στο δεξί). Στην περίπτωση αυτή, το πλήθος των στοιχείων της κάθε ομάδας θα διαμορφωθεί ως εξής: $|α| = 2k – 2$, $|β| = |γ| = k – 1$ και $|δ| = 2$.
ii. Όταν έχουμε n = 4k + 1, θα επιλέξουμε την μορφή Β, όπου στο πρώτο ζύγισμα και τα δύο κανονικά νομίσματα βρίσκονται στο ίδιο μέρος της ζυγαριάς (τα 0,1 βρίσκονται στο αριστερό μέρος της ζυγαριάς). Στην περίπτωση αυτή, το πλήθος των στοιχείων της κάθε ομάδα θα διαμορφωθεί ως εξής: $|α| = 2k – 2$, $|β| = |γ| = k$ και $|δ| = 1$.
iii. Όταν έχουμε n = 4k + 2, θα επιλέξουμε την μορφή Β και το πλήθος των στοιχείων της κάθε ομάδας θα διαμορφωθεί ως εξής: $|α| = 2k – 2$, $|β| = |γ| = k$ και $|δ| = 2$.
iv. Όταν έχουμε n = 4k + 3, θα επιλέξουμε την μορφή Α και το πλήθος των στοιχείων της κάθε ομάδας θα διαμορφωθεί ως εξής: $|α| = 2k$, $|β| = |γ| = k$ και $|δ| = 1$.

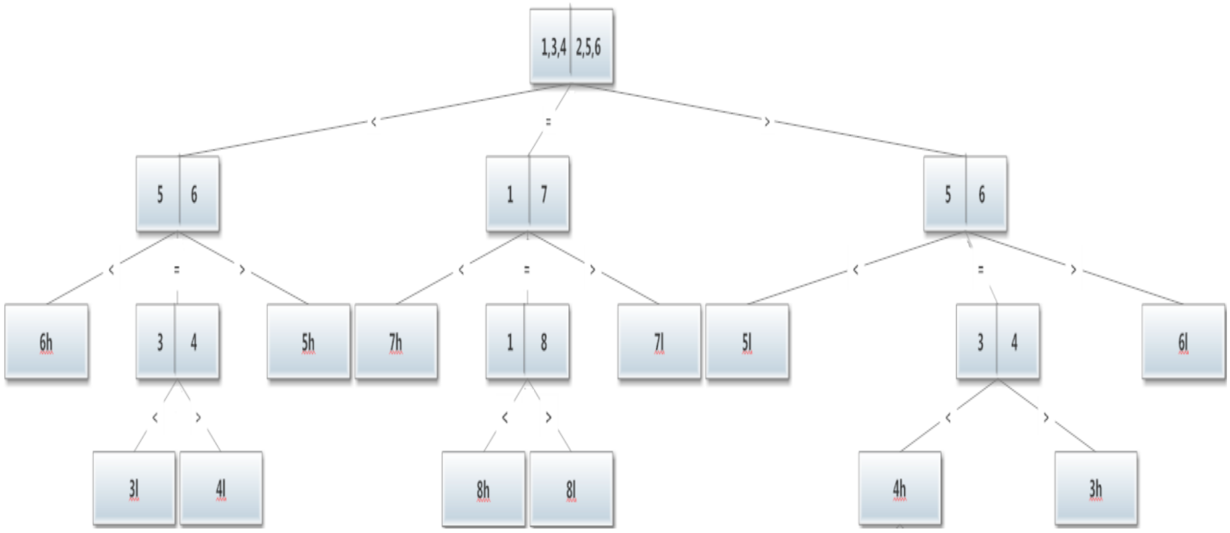
Παρατηρούμε και στις τέσσερις περιπτώσεις ο πληθάριθμος της ομάδας β είναι ίσος με τον πληθάριθμο της ομάδας γ καθώς και ότι η ομάδα δ αποτελείται είτε από ένα είτε από δύο στοιχεία.

Η παραπάνω διαδικασία μας οδηγεί σε υποπεριπτώσεις με σύγκριση (ζύγιση) πολύ λιγότερων νομισμάτων ($|β| + |γ| \leq 2κ$ σε πλήθος, αφού η περίπτωση της αρχικής ισότητας, που οδηγεί στην ομάδα δ είναι τετριμμένη).
Έτσι, για παράδειγμα, για τις περιπτώσεις n = 7,8,…,14 έχουμε τα εξής:

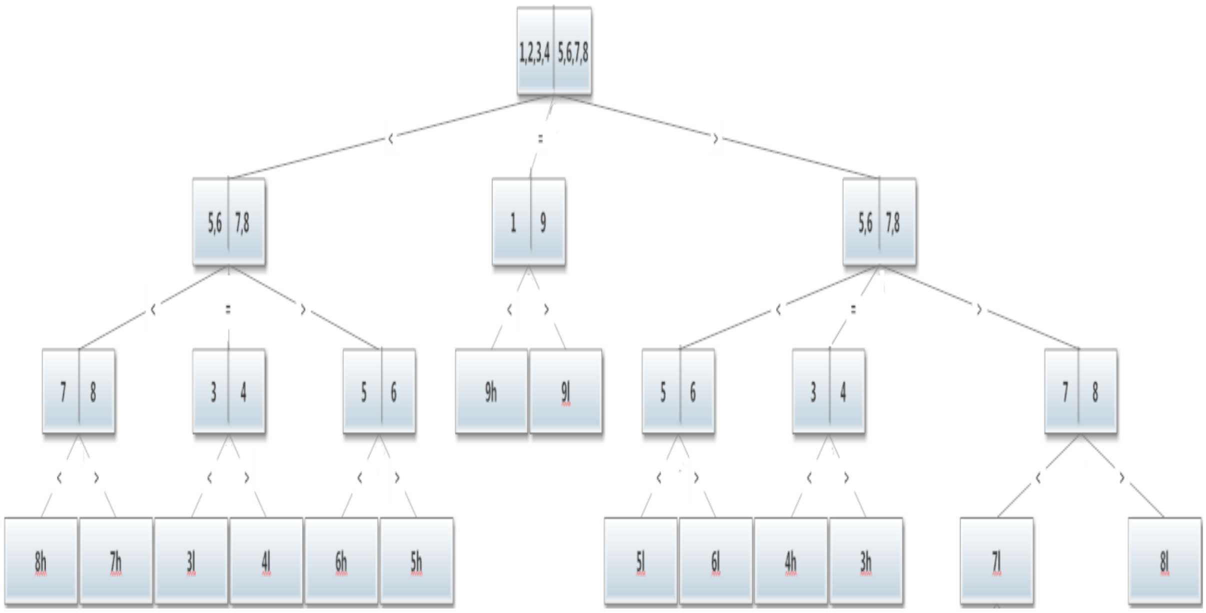
- Όταν έχουμε n = 7 είμαστε στην περίπτωση (iv). Συνεπώς n = 4k + 3 με k = 1, και $|α| = 2*k = 2*1 = 2$, $|β| = |γ| = k = 1$ και $|δ| =1$ οπότε θα προκύψουν οι ακόλουθες ομάδες: α = {2,3}, β = {4}, γ = {5} και δ = {6}. Έχουμε:



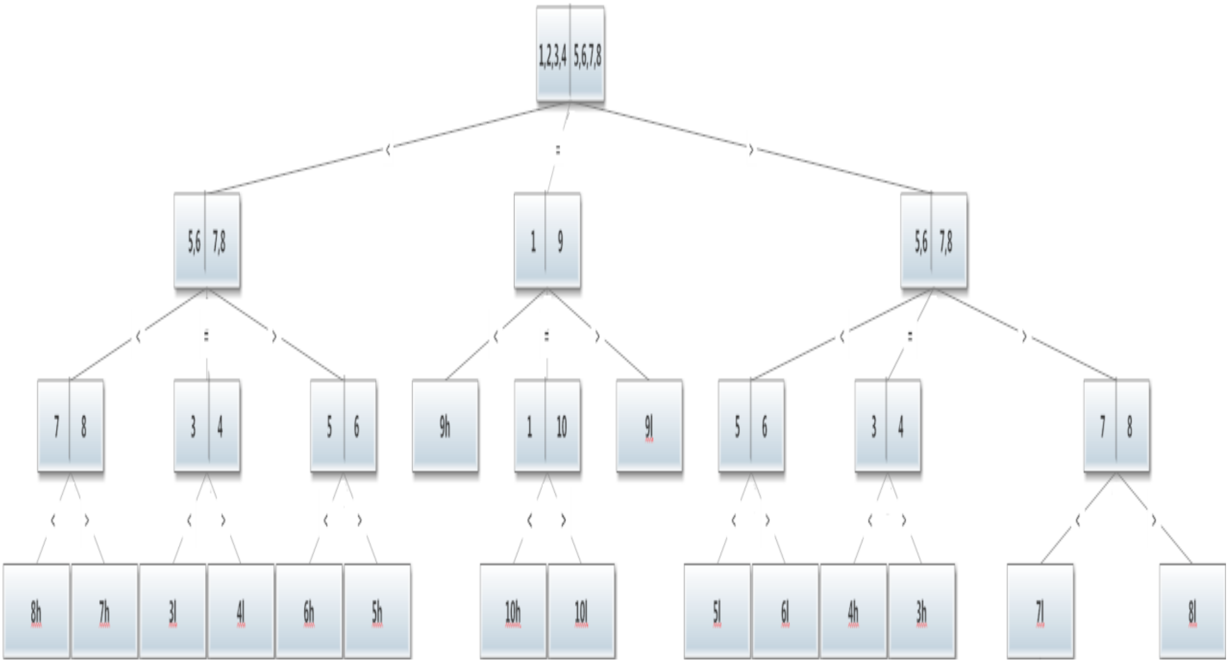
- Όταν έχουμε n = 8 είμαστε στην περίπτωση (i). Άρα n = 4k με k =2, και |α| = 2*k – 2 = 2*2 – 2 = 2, |β| = |γ| = k –1 = 2 –1 = 1 και |δ| = 2 οπότε θα προκύψουν οι ακόλουθες ομάδες: α = {2,3}, β = {4}, γ = {5} και δ = {6,7}. Έχουμε:



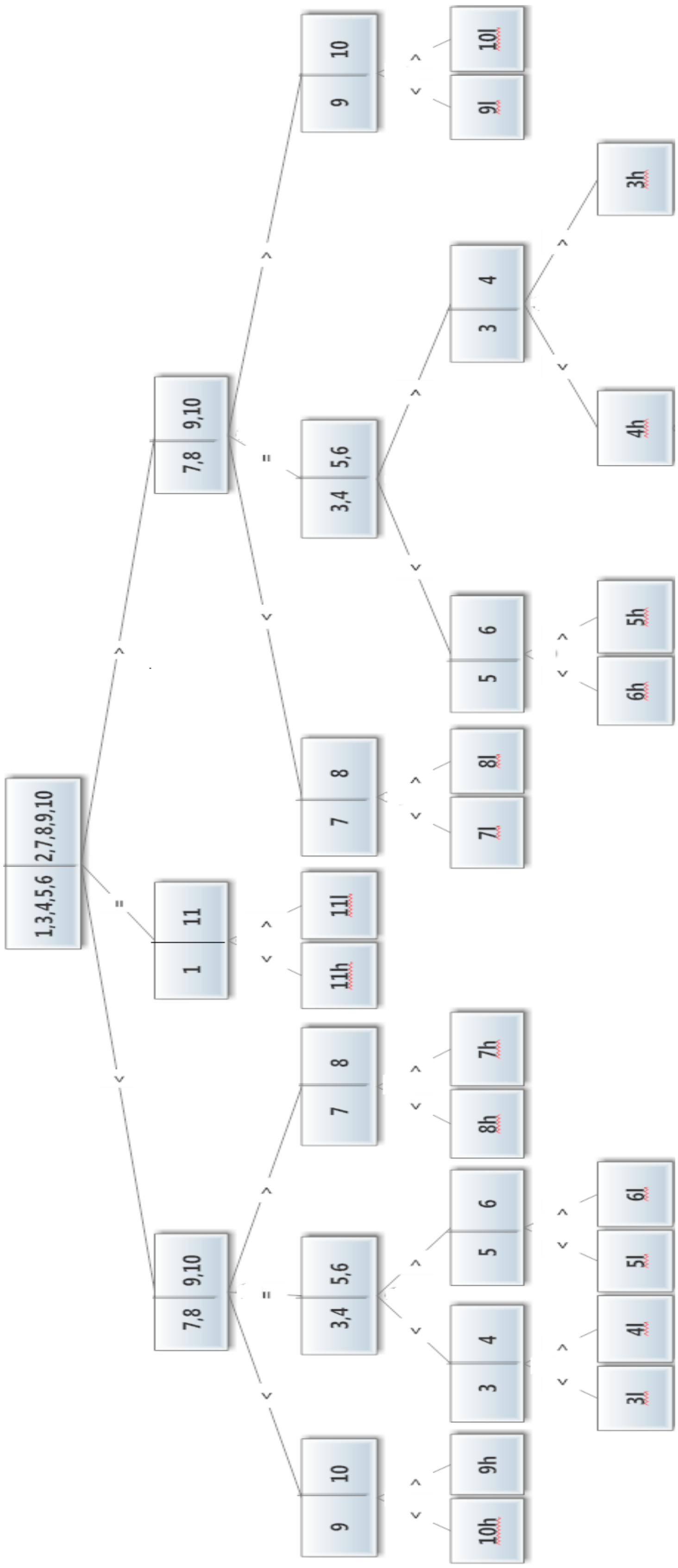
- Όταν έχουμε n = 9 είμαστε στην περίπτωση (ii). Συνεπώς n = 4k + 1 με k = 2, και |α| = 2*k – 2 = 2*2 – 2 = 2, |β| = |γ| = k = 2 και |δ| = 1 οπότε θα προκύψουν οι ακόλουθες ομάδες: α = {2,3}, β = {4,5}, γ = {6,7} και δ = {8}. Έχουμε:



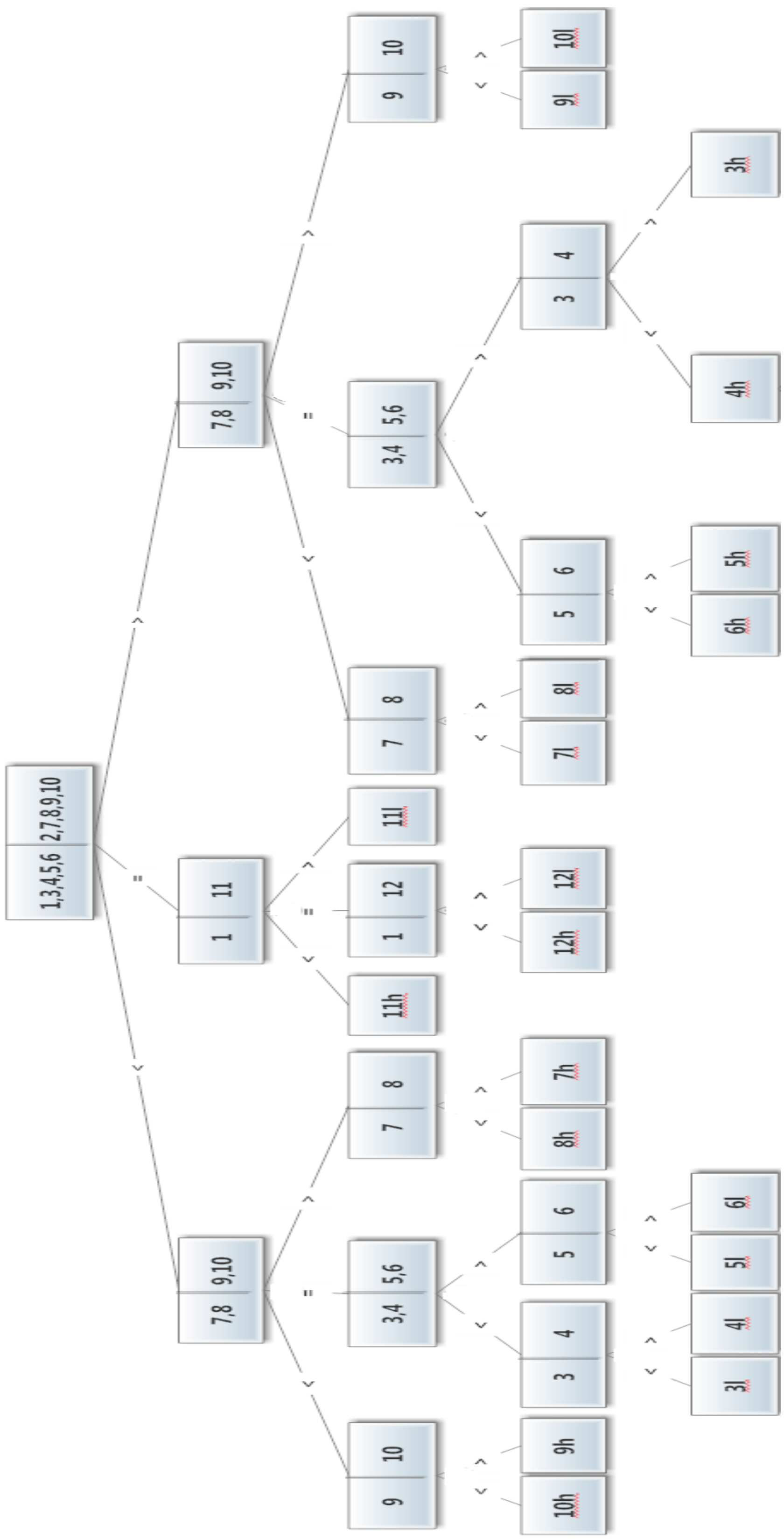
- Όταν έχουμε n = 10 είμαστε στην περίπτωση (iii). Συνεπώς n = 4*k + 2 με k =2, και|α| = 2k – 2 = 2*2 – 2 = 2,|β| =|γ| = k = 2 και|δ| = 2 οπότε θα προκύψουν οι ακόλουθες ομάδες: α = {2,3}, β = {4,5}, γ = {6,7} και δ = {8,9}. Έχουμε:



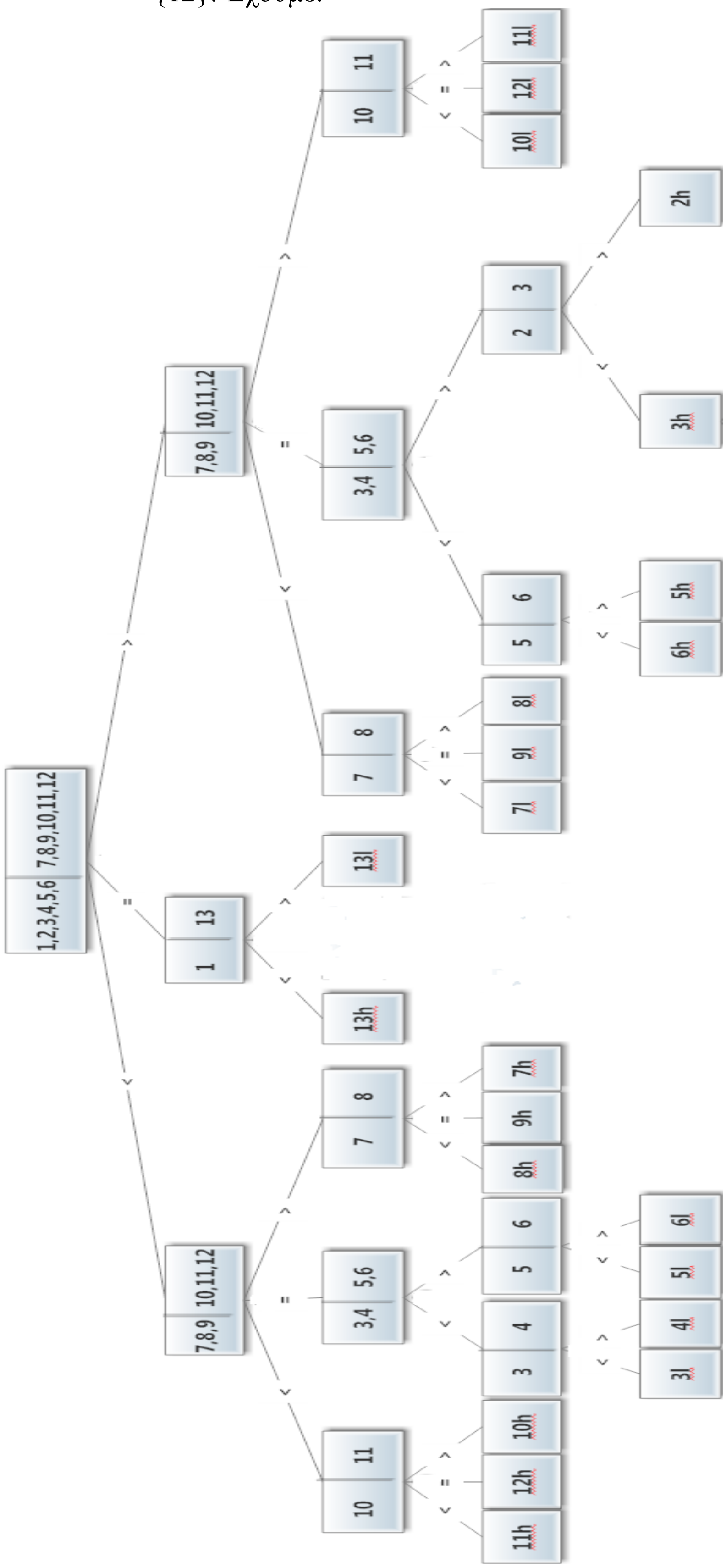
- Όταν έχουμε n = 11 είμαστε στην περίπτωση (iv). Επομένως n = 4k + 3 με k = 2, και|α| = 2k = 2*2 = 4,|β| =|γ| = k = 2 και|δ| = 1 οπότε θα προκύψουν οι ακόλουθες ομάδες: α = {2,3,4,5}, β = {6,7}, γ = {8,9} και δ = {10}. Έχουμε:



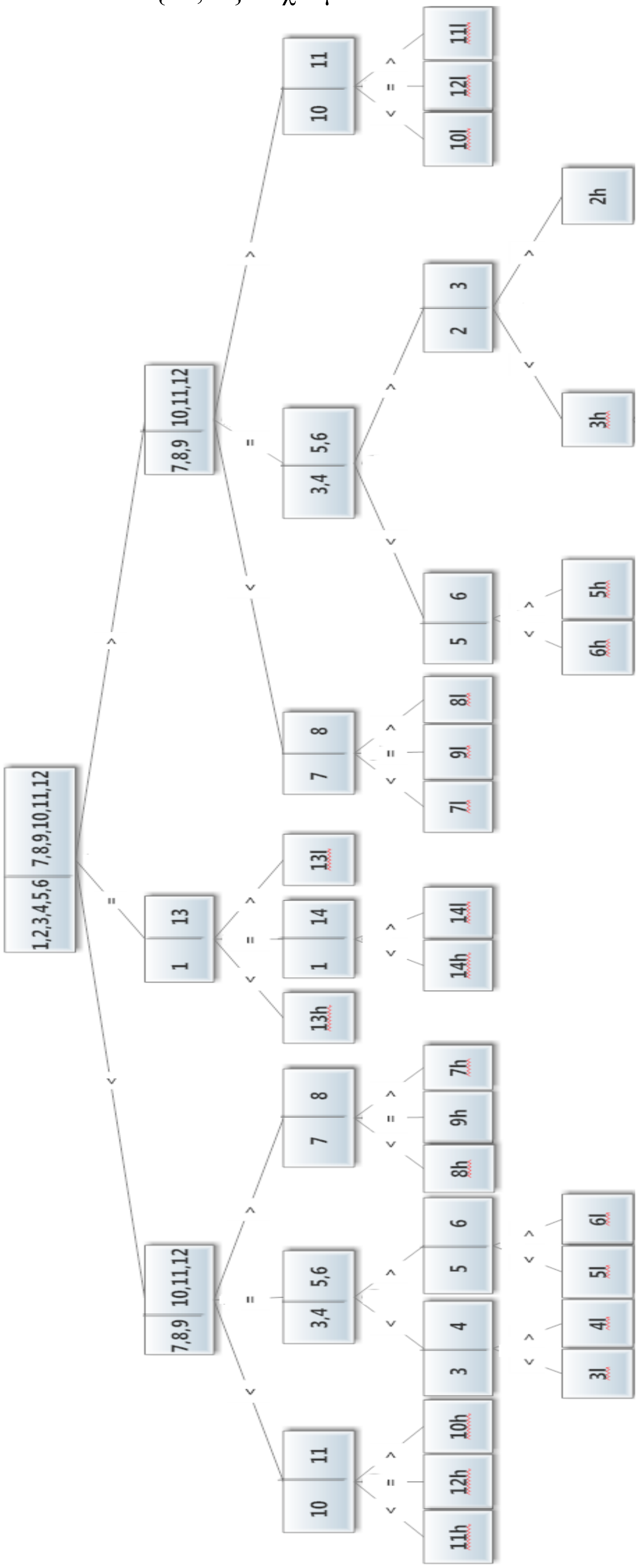
- Όταν έχουμε n = 12 είμαστε στην περίπτωση (i). Συνεπώς, n = 4k με k = 3, και|α| = 2k – 2 = 2*3 – 2 = 4,|β| =|γ| = k – 1 και|δ| = 2 οπότε θα προκύψουν οι εξής ομάδες: α = {2,3,4,5}, β = {6,7}, γ = {8,9} και δ = {10,11}. Έχουμε:



- Όταν έχουμε $n = 13$ είμαστε στην περίπτωση (ii). Άρα $n = 4k + 1$ με $k = 3$, και $|α| = 2k – 2 = 2*3 – 2 = 4$, $|β| = |γ| = k = 3$ και $|δ| = 1$ οπότε θα προκύψουν οι εξής ομάδες: α = {2,3,4,5}, β = {6,7,8}, γ = {9,10,11} και δ = {12}. Έχουμε:



- Όταν έχουμε n = 14 είμαστε στην περίπτωση (iii). Άρα n = 4k +2 με k = 3, και|α| = 2k – 2 = 2*3 – 2 = 4, |β| = |γ| = k = 3 και|δ| = 2 οπότε θα προκύψουν οι εξής ομάδες: α = {2,3,4,5}, β = {6,7,8}, γ = {9,10,11} και δ = {12,13}. Έχουμε:



Βιβλιογραφία

[1] I. Bosnjak, Some New Results Concerning Three Counterfeit Coins Problem, *Discrete Applied Mathematics* **48**, pp 81-85, 1994.

[2] G. Chang, F. Hwang and S. Lin, Group Testing With Two Defectives, *Discrete Applied Mathematics* **4**, pp 97-102, 1982.

[3] J. Ghosh, S.K. Ghosh and R.K. Pal, A Revisit to the Eight Coins Problem, *International Journal of Computing and Information Technology*, vol. 2, no. **1**, pp 1-14, 2010.

[4] J. Ghosh, P. Senmajumdar, S. Maitra, D. Dhal and R.K. Pal, A Generalized Algorithm for Solving n Coins Problem, Computer Science and Automation Engineering International Conference on, Vol **2**, pp 411-415.

[5] F. Hwang, A Tale of Two Coins, *Amer. Math. Monthly* **2**, pp 121-129, 1987.

[6] A. Li, Three Counterfeit Coins Problem, *Journal of Combinatorial Theory*, Series A **66**, pp 93-101, 1994.

[7] B. Manvel, Counterfeit Coin Problems, *Math Mag.* **50**, pp 90-92, 1977.

[8] L. Pyber, How To Find Many Counterfeit Coins?, *Graphs and Combinatorics* ***2***, pp 173-177, 1986.

[9] M. Qi, S. Liu, Selecting Two Different Defective Coins, *Applied Mathematics and Computation* **180**, pp 559-568, 2006.

[10] K.H. Rosen, *Discrete Mathematics and Its Applications* (Fifth Edition), Tata McGraw-Hill Publishing Co. Ltd, New Delhi, 2003.

[11] R. Tosic, Two Counterfeit Coins, *Discrete Math.* **46**, pp 295-298, 1983.

[12] L.Wen-An and N. Zan-Kan, Optimal Detection of Two Counterfeit Coins With Two-Arms Balance, *Discrete Applied Mathematics* **137**, pp 267-291, 2004.

Παράρτημα

Παρακάτω ακολουθεί ένα πρόγραμμα οπού υλοποιεί την περίπτωση κατά την οποία έχουμε n νομίσματα ένα εκ των οποίων είναι κίβδηλο και γνωρίζουμε ότι είναι ελαφρύτερο από τα υπόλοιπα. Κατά την εκτέλεση του προγράμματος, σε κάθε βήμα (ζύγισμα) βλέπουμε ποια νομίσματα τοποθετούνται σε κάθε ένα από τα δύο σκέλη της ζυγαριάς καθώς και ποιο από τα δύο σκέλη της είναι το ελαφρύτερο. Κάθε φορά αναζητούμε το κίβδηλο νόμισμα στο ελαφρύτερο σκέλος της ζυγαριάς. Μετά από έναν αριθμό διαδοχικών νομισμάτων φτάνουμε στη λύση του προβλήματος όπου εμφανίζεται το κίβδηλο νόμισμα καθώς και ο συνολικός αριθμός των ζυγισμάτων που έγιναν μέχρι να βρεθεί η λύση.

```
package coinproblem;

import java.util.ArrayList;
import java.util.Random;

/**
 *
 * @author user
 */
public class CoinProblem {



    public static int input(String s) {
        byte buffer[] = new byte[100];
        System.out.println(s + ":");
        try {
            System.in.read(buffer);
        } catch (java.io.IOException e) {
            System.out.println("Exception encountered:" + e.getMessage());
            System.exit(1);
        }
        return (new Integer((new String(buffer)))).intValue();
    }

    // Επιστρέφει -1 αν το βάρος των νομισμάτων της αριστερής λίστας είναι
μικρότερο
    // από το βάρος των νομισμάτων της δεξιάς λίστας
    // Επιστρέφει 0 αν τα βάρη είναι ίσα.
    // Επιστρέφει 1 αν η αριστερή λίστα είναι βαρύτερη από τη δεξιά.
    public static int scale(int[] coinWeights, ArrayList<Integer> leftList,
ArrayList<Integer> rightList){
        int val = 0;
        int leftWeight = 0;
        int rightWeight = 0;
        for(int j=0; j<leftList.size(); j++){
```

```
      leftWeight+=coinWeights[leftList.get(j)-1];
    }
    for(int j=0; j<rightList.size(); j++){
      rightWeight+=coinWeights[rightList.get(j)-1];
    }

    if(leftWeight<rightWeight){ val=-1; }
    if(leftWeight==rightWeight){ val=0; }
    if(leftWeight>rightWeight){ val=1; }
    return val;
  }

  public static ArrayList<Integer> createCoinNames(int n){
    ArrayList<Integer> coins = new ArrayList<Integer>();
    for(int j=1; j<=n; j++){
      coins.add(j);
    }
    return coins;
  }

  public static int[] createCoinWeights(int n){
    int[] coinWeights = new int[n];
    for(int j=0; j<n; j++){
      coinWeights[j]=10;
    }
    Random pos = new Random();
    Random defectiveWeight = new Random();
    coinWeights[pos.nextInt(n)]=defectiveWeight.nextInt(1*10)+1;
    return coinWeights;
  }



  public static int findSplitSize(int size){
    int val = 3;
    boolean isMultipleof3 = false;

    while(val<=size){
      if(val == size) return size/3;
      val *= 3;
    }
    return size/3 +1;
  }


  public static ArrayList<Integer> subList(ArrayList<Integer> list, int start, int end){
    ArrayList<Integer> sublist;
    sublist = new ArrayList<Integer>(list.subList(start, end));
    return sublist;
  }
```

```
    public static void printArray2(int[] a){
        System.out.print("[");
        for(int j=0; j<a.length-1; j++){
            System.out.print(a[j]+"\t");
        }
        System.out.print(a[a.length-1]);
        System.out.print("]");
    }

    public static void printArray(int[] a){
        System.out.print("[");
        for(int j=0; j<a.length-1; j++){
            System.out.print(a[j]+", ");
        }
        System.out.print(a[a.length-1]);
        System.out.print("]");
    }

    public static void printArrayList(ArrayList<Integer> a){
        System.out.print("[");
        for(int j=0; j<a.size()-1; j++){
            System.out.print(a.get(j) +"\t");
        }
        System.out.print(a.get(a.size()-1));
        System.out.print("]");
    }



    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        //System.out.println(CreateCoinWeights(10));
        int n = 10;
        int[] coinWeights = createCoinWeights(n);
        ArrayList<Integer> theList = createCoinNames(n);
        ArrayList<Integer> leftlist = new ArrayList<Integer>();
        ArrayList<Integer> rightlist = new ArrayList<Integer>();
        ArrayList<Integer> middlelist = new ArrayList<Integer>();

        System.out.println("Αρχική λίστα νομισμάτων: " + theList);
        //System.out.println("Initial coin list: "); printArrayList(theList);
System.out.println("");
        System.out.print("(Μυστικά) βάρη νομισμάτων: "); printArray(coinWeights);
System.out.println("");
        int counter = 0;
        while(theList.size()>1){
            counter++;
            System.out.println("******* Ζύγισμα "+counter);
```

```
        int splitsize = findSplitSize(theList.size());
        leftlist = subList(theList, 0, splitsize);
        System.out.println("Στο αριστερό σκέλος της ζυγαριάς τοποθετούνται τα 
νομίσματα: " + leftlist);
        rightlist = subList(theList, splitsize, 2*splitsize);
        System.out.println("Στο δεξί σκέλος της ζυγαριάς τοποθετούνται τα 
νομίσματα: " + rightlist);
        middlelist = subList(theList, 2*splitsize, theList.size());
        System.out.println("Στην άκρη κρατάμε τα νομίσματα: " + middlelist);
        if(scale(coinWeights, leftlist, rightlist)==-1){
          System.out.println("Το αριστερό σκέλος είναι ελαφρύτερο");
          theList = leftlist;
        }
        if(scale(coinWeights, leftlist, rightlist)==0){
          System.out.println("Τα δύο σκέλη έχουν ίσο βάρος");
          theList = middlelist;

        }
        if(scale(coinWeights, leftlist, rightlist)==1){
          System.out.println("Το δεξί σκέλος είναι ελαφρύτερο");
          theList = rightlist;

        }
        System.out.println("Άρα, το κίβδηλο νόμισμα βρίσκεται μεταξύ των 
νομισμάτων: " + theList);

      }
      System.out.println("******* Τελική απάντηση: ");
      if(!theList.isEmpty()){
        System.out.println("Το κίβδηλο νόμισμα είναι το νόμισμα: " + theList.get(0));
      }
      if(theList.isEmpty()){
        System.out.println("Όλα τα νομίσματα είναι γνήσια");
      }
      System.out.println("Για την απάντηση χρειάστηκαν: "+counter+" ζυγίσματα.");

      }
    }
```