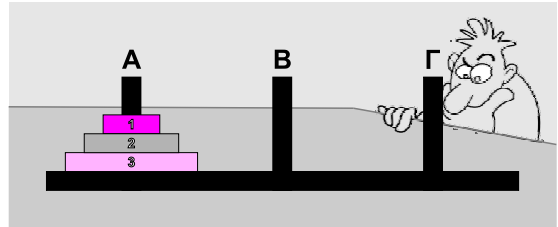


## 5η Εργασία – Πύργοι του Ανόι

### Ανάλυση του προβλήματος

Θα προσπαθήσουμε να αναλύσουμε τη διαδικασία με την παραδοχή ότι υπάρχουν τρεις δίσκοι στον στύλο A.

Ας υποθέσουμε ότι θέλουμε να μετακινήσουμε μόνο έναν δίσκο, τον δίσκο 1, από τον στύλο A στον στύλο Γ. Η λύση είναι απλή και περιλαμβάνει μόνο μια κίνηση, χωρίς να χρησιμοποιείται ο βοηθητικός στύλος B.



Αν τώρα θέλουμε να μετακινήσουμε τους δύο δίσκους 1 και 2, από τον στύλο A στον στύλο Γ, θα χρειαστούν τρεις κινήσεις. Έτσι, θα μετακινήσουμε τον δίσκο 1 στον στύλο B, μετά τον δίσκο 2 στον στύλο Γ, και τέλος τον δίσκο 2 από τον στύλο B στον Γ.

Ας δούμε τώρα πώς θα μπορέσουμε να μετακινήσουμε και τους τρεις δίσκους από τον στύλο A στον στύλο Γ. Αν μετακινήσουμε τους δύο δίσκους 1 και 2 από τον στύλο A στον B, θα μείνει μόνο ο δίσκος 3 στον στύλο A. Μετακινούμε τώρα τον δίσκο 3 από τον στύλο A στον Γ και μετά τους δύο δίσκους 1 και 2 από τον στύλο B στον Γ. Παρατηρούμε επομένως ότι το πρόβλημα της μετακίνησης των τριών δίσκων ανάγεται σε πρόβλημα μετακίνησης δύο δίσκων, το οποίο με τη σειρά του ανάγεται σε πρόβλημα μετακίνησης μόνο ενός δίσκου. Αυτή είναι μια αναδρομική διαδικασία η οποία συνοψίζεται στις παρακάτω προτάσεις:

Αν θέλεις να μετακινήσεις  $N$  δίσκους από τον στύλο A στον Γ, χρησιμοποιώντας ως βοηθητικό τον στύλο B, ακολούθησε τα παρακάτω βήματα:

1. Μετακίνησε  $N-1$  δίσκους από τον A στον B με βοηθητικό τον στύλο Γ
2. Μετακίνησε τον  $N$  δίσκο από τον στύλο A στον στύλο Γ
3. Μετακίνησε  $N-1$  δίσκους από τον B στον Γ με βοηθητικό τον στύλο A

### Βήματα

- ❶ Το βασικότερο βήμα είναι η δημιουργία της αναδρομικής συνάρτησης `towers()`, η οποία θα δέχεται ως ορίσματα το πλήθος των δίσκων που θα μετακινήσουμε, τον στύλο από τον οποίο θα μετακινηθούν, τον στύλο στον οποίο θα καταλήξουν, και τον βοηθητικό στύλο που μπορεί να χρησιμοποιηθεί.
- ❷ Η συνάρτηση `towers()` αρχικά καλεί αναδρομικά τον εαυτό της για να μετακινήσει `num-1` δίσκους από τον αρχικό στύλο στον βοηθητικό.
- ❸ Αμέσως μετά μετακινείται ο δίσκος `num` στον προορισμό του.
- ❹ Στη συνέχεια η συνάρτηση `towers()` καλεί πάλι αναδρομικά τον εαυτό της για να μετακινήσει τους `num-1` δίσκους από τον βοηθητικό στύλο στον τελικό προορισμό τους.
- ❺ Στην περίπτωση που η συνάρτηση `towers()` κληθεί με πλήθος 1, τότε απλώς μετακινείται ο δίσκος χωρίς αναδρομικές κλήσεις της συνάρτησης.
- ❻ Η κλήση της συνάρτησης `towers()` γίνεται από τη `main()` ώστε να μετακινήσει `n` πλήθος δίσκων από τον στύλο A στον C με βοηθητικό στύλο τον B. Το πλήθος `n` των δίσκων δίνεται από τον χρήστη.

## Κώδικας

```
#include <stdio.h>
#include <math.h>
void towers(int num, char stylos_apo, char stylos_se, char stylos_temp);
int main(void)
{
    int n,ar;
    printf("Δώσε αριθμό δίσκων : ");
    scanf("%d", &n);
    printf("Θα χρειαστούν οι ακόλουθες κινήσεις:\n\n");
    towers(n, 'A', 'C', 'B');
    ar=pow(2,n)-1;
    printf("\nΧρειάστηκαν %d μετακινήσεις\n",ar);
    return 0;
}

void towers(int num, char stylos_apo, char stylos_se, char stylos_temp)
{
    if (num == 1)
    {
        printf("Μετακίνησε τον δίσκο 1 από τον στύλο %c στον στύλο %c\n",stylos_apo,
            stylos_se);
        return;
    }
    towers(num-1,stylos_apo,stylos_temp,stylos_se);
    printf("Μετακίνησε τον δίσκο %d από τον στύλο %c στον στύλο %c\n",num,stylos_apo,
        stylos_se);
    towers(num-1,stylos_temp,stylos_se,stylos_apo);
}
```

e5.c

❶ Κλήση της συνάρτησης **towers()** για τη μετακίνηση **n** δίσκων από τον στύλο **A** στον **C** με βοηθητικό τον **B**.

❶ Αναδρομική συνάρτηση στην οποία μεταβιβάζονται το πλήθος των προς μετακίνηση δίσκων (**num**), ο στύλος από τον οποίον θα τους πάρουμε (**stylos\_apo**), ο στύλος στον οποίο θα καταλήξουν (**stylos\_se**) και ο βοηθητικός στύλος που μπορεί να χρησιμοποιηθεί (**stylos\_temp**).

❷ Μη αναδρομική περίπτωση.

❷ Η συνάρτηση **towers()** καλείται αναδρομικά με μικρότερο κατά ένα πλήθος δίσκων (**num-1**). Μετακινεί **num-1** δίσκους από τον αρχικό στύλο στον βοηθητικό.

❸ Μετακινείται ο δίσκος **num**.

❸ Η συνάρτηση **towers()** καλείται αναδρομικά για να μετακινήσει τους **num-1** δίσκους από τον βοηθητικό στον τελικό προορισμό τους.

## Προτάσεις

Μια παραλλαγή αυτού του κλασικού προβλήματος είναι οι κυκλικοί πύργοι του Ανόι (Cyclic Hanoi), όπου οι στύλοι A, B και Γ βρίσκονται κυκλικά τοποθετημένοι, σύμφωνα με τη φορά των δεικτών του ρολογιού. Ο επιπλέον περιορισμός που τίθεται είναι ότι οι μετακινήσεις δίσκων μπορούν να γίνουν μόνο κατά τη φορά των δεικτών του ρολογιού. Δηλαδή, από τον A στον B, ή από τον B στον Γ, ή από τον Γ στον A. Αν θελήσετε να υλοποιήσετε αυτή την παραλλαγή σας εύχομαι καλό κουράγιο!