

Παραδείγματα Ασκήσεων Δομών Δεδομένων

Άσκηση 1: Να δοθεί ο τυπικός ορισμός του ΑΤΔ Σύνολο.

Απάντηση:

Το ΑΤΔ σύνολο ορίζεται ως μία συλλογή μοναδικών στοιχείων του ίδιου τύπου, τα οποία δεν έχουν καμία σχέση μεταξύ τους.

Άσκηση 2: Να δοθεί η υλοποίηση ΑΤΔ στοίβα με πίνακα.

Απάντηση:

```
#define StackLimit 50 /*το όριο μεγέθους της στοίβας*/
```

```
typedef int StackElementType;
```

```
typedef struct {  
    int Top;  
    StackElementType Element[StackLimit];  
} StackType;
```

```
typedef enum {  
    FALSE, TRUE  
} boolean;
```

Άσκηση 3: Να δοθεί η υλοποίηση της εισαγωγής στοιχείου του ΑΤΔ ουρά με Αλγόριθμο ή κώδικα.

Απάντηση:

Α' τρόπος με αλγόριθμο:

AddQ

*/*Αλγόριθμος εισαγωγής ενός στοιχείου Item στην πίσω άκρη της ουράς Queue, εφόσον ο πίνακας Element δεν είναι γεμάτος*/*

1. $NewRear \leftarrow (Queue.Rear + 1) \bmod QueueLimit$

2. Αν $NewRear \neq Queue.Front$ τότε

α. $QueueError \leftarrow false$

β. $Queue.Element[Queue.Rear] \leftarrow Item$

γ. $Queue.Rear \leftarrow NewRear$

Αλλιώς

α. Γράψε 'Προέκυψε σφάλμα ουράς: γεμάτη ουρά'

β. $QueueError \leftarrow true$

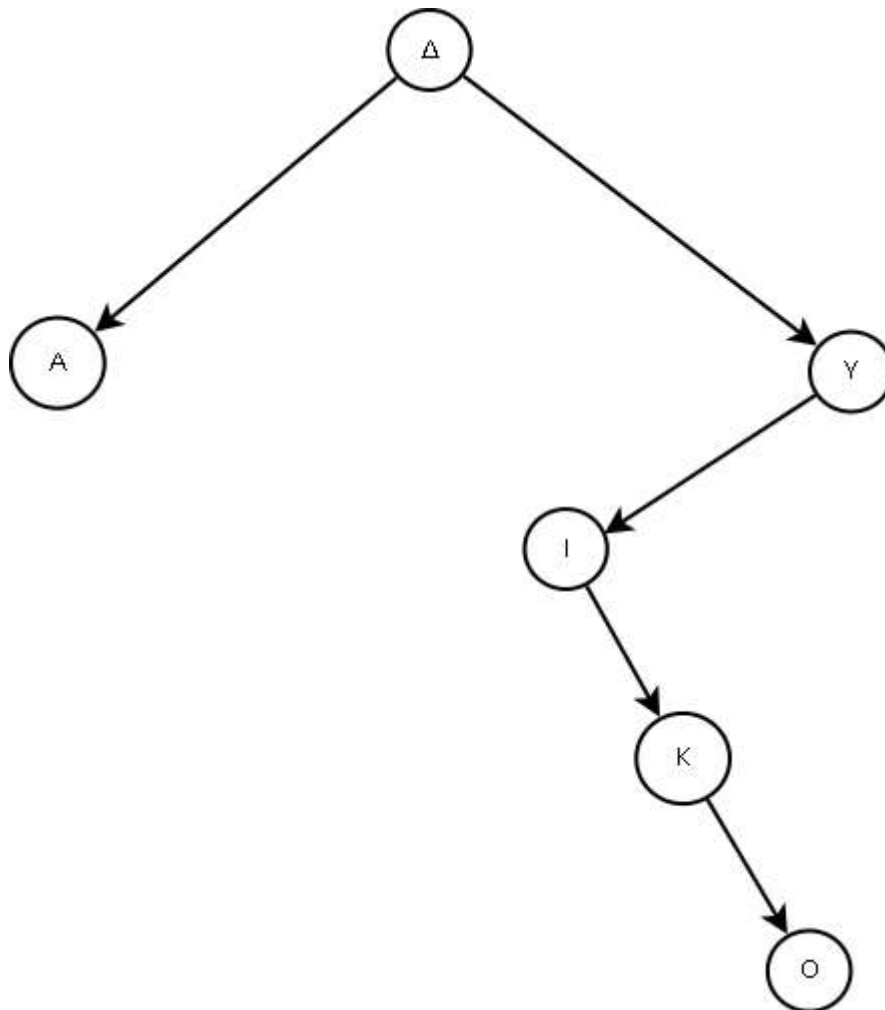
Τέλος_αν

B' τρόποσ σε C:

```
void AddQ(QueueType *Queue, QueueElementType Item)
{
    int NewRear;
    if(!FullQ (*Queue))
    {
        NewRear = (Queue ->Rear + 1) % QueueLimit;
        Queue ->Element[Queue ->Rear] = Item;
        Queue ->Rear = NewRear;
    }
    else
        printf("Full Queue\n");
}
```

Άσκηση 4: Κατασκευάστε το δυαδικό δέντρο αναζήτησης (ΔΔΑ) που προκύπτει από τα γράμματα της λέξης «ΔΥΑΔΙΚΟ».

Απάντηση:



Άσκηση 5: Διασχίσετε το ΔΔΑ της Άσκησης 4 με α) ενδοδιατεταγμένη, β)μεταδιατεταγμένη και γ) προδιατεταγμένη διάσχιση.

Απάντηση:

α) ΑΔΙΚΟΥ

β) ΑΟΚΙΥΔ

γ)ΔΑΥΙΚΟ