

```
typedef ??? ListElementType; //τύπος για τους μαθητές που θα εισαχθούν στη λίστα
```

```
typedef ??? StackElementType; //τύπος για τους ΑΜ που θα εισαχθούν στη στοίβα
```

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

```
float FindMins(ListPointer List, NodeType Node[NumberOfNodes], StackType *Stack);
```

```
int main() {  
    Δηλώσεις μεταβλητών  
    ...  
  
    //A  
    Αρχικοποίηση της δεξαμενή κόμβων (InitializeStoragePool(...))  
  
    //B  
    Δημιουργία της λίστας (CreateList(...))  
  
    //C  
    Εμφάνιση της δεξαμενής κόμβων (printALL(...))  
  
    //D  
    Εμφάνιση της λίστας (TraverseLinked(...))  
  
    //E  
    Με επαναληπτικό βρόχο (for)  
        Διάβασε τον αριθμό μητρώου του μαθητή  
        Διάβασε τον βαθμό του μαθητή  
        //Εισαγωγή του μαθητή στην αρχή της λίστας  
        Εισαγωγή του μαθητή στη λίστα (Insert(...))  
  
    //F  
    Εμφάνιση της δεξαμενής κόμβων  
  
    //G  
    Εμφάνιση της λίστας
```

//H

Αν η λίστα είναι κενή

Εμφάνισε "Empty List"

Αλλιώς

Εμφάνισε "Not an Empty List"

//I

Αν η λίστα είναι γεμάτη

Εμφάνισε "Full List"

Αλλιώς

Εμφάνισε "Not a Full List"

//J

Εύρεση του ελάχιστου βαθμού και ανάθεση στο min (**FindMins(...)**)

Εμφάνιση του ελάχιστου βαθμού

//Εμφάνιση των AM που έχουν τον ελάχιστο βαθμό

Όσο η στοίβα δεν είναι κενή

Απώθησε τον AM από την κορυφή της στοίβας

Εμφάνισε τον AM

//K

Εμφάνισε τη στοίβα (**TraverseStack(...)**)

//L

Εμφάνιση της δεξαμενής κόμβων

//M

Εμφάνιση της λίστας

```
float FindMins(ListPointer List, NodeType Node[NumberOfNodes], StackType *Stack)
```

```
{  
    Δηλώνω τις μεταβλητές  
    _____ CurrPtr;  
    _____ min;
```

Δημιουργία στοίβας Stack (**CreateStack(...)**)

Αν η List δεν είναι άδεια{

Αρχικοποίηση του αριθμοδείκτη CurrPtr να δείχνει το 1^ο στοιχείο της λίστας

Αρχικοποίηση της min στον βαθμό του πρώτου στοιχείου της λίστας

Όσο δεν έχω φτάσει στο τέλος της λίστας{

Αν ο βαθμός του τρέχοντος στοιχείου της λίστας <= min {

Αν ο βαθμός του τρέχοντος στοιχείου της λίστας < min {

//Βρέθηκε μικρότερος βαθμός, η στοίβα πρέπει να αδειάσει για να

//εισαχθεί ο νέος AM που αντιστοιχεί στο νέο min

Όσο η στοίβα δεν είναι κενή{

Απώθησε το κορυφαίο στοιχείο της στοίβας

}

Ανάθεση του βαθμού του τρέχοντος στοιχείου στο min

}

Ωθησε το AM του τρέχοντος στοιχείου στη στοίβα

}

Προχώρησε στο επόμενο στοιχείο της λίστας

//ενημέρωση τιμής CurrPtr

}

}

Αλλιώς

Εμφάνισε “ Empty List ...”

Επιστροφή της min

}