

ΨΗΦΙΑΚΟΙ ΥΠΟΛΟΓΙΣΤΕΣ

ΑΝΑΦΟΡΑ ΕΡΓΑΣΤΗΡΙΟΥ 2

ΜΙΧΑΛΗΣ ΓΑΛΑΝΗΣ 2016030036

ΓΙΩΡΓΟΣ ΒΙΡΑΚΗΣ 2016030035

Προεργασία

Η δεύτερη άσκηση ζητούσε να κατασκευάσουμε ένα πρόγραμμα σε C και μετά να το μετατρέψουμε στη γλώσσα Clang. Αυτό προαπαιτούσε να έχουμε εξοικειωθεί με τη C και να έχουμε κατανοήσει τα βασικά χαρακτηριστικά της Clang.

Περιγραφή Ζητούμενων

Σκοπός αυτού του εργαστηρίου ήταν η εξοικείωση μας με τη γλώσσα Clang.

Ζητηθήκαμε να κατασκευάσουμε ένα πρόγραμμα σε γλώσσα C που είχε ως ασχολία βασική διαχείριση λιστών (προσθήκη στοιχείων, αναζήτηση, διαγραφή, εκτύπωση). Μας ζητήθηκε αργότερα να μετατρέψουμε τις πρώτες 5 λειτουργίες του προγράμματος σε Clang.

Περιγραφή της Εκτέλεσης

Γενικά η ανάπτυξη του προγράμματος σε Clang έμοιαζε πολύ με αυτή της C. Οι συναρτήσεις οριζόντουσαν με τον ίδιο τρόπο και η δομή του προγράμματος ήταν η ίδια (υπήρχε η `main` που καλούσε διάφορες συναρτήσεις και το αποτελεσμα τους επέστρεφε σε αυτή). Όμως, για να προσωμοιώσουμε γλώσσα «χαμηλότερου επιπέδου»:

- Όλες οι μεταβλητές αποθηκευόντουσαν ως integers σε καταχωρητές (registers), προκειμένου να πραγματοποιηθούν αριθμητικές και λογικές πράξεις.
- Λόγω έλλειψης των εντολών συνθήκης και επανάληψης, η διακοπή ροής στο πρόγραμμα γινόταν με σωστή εφαρμογή των εντολών **label** και **goto**.

Σημείωση: επειδή κάποιες συναρτήσεις είχαν κοινό τμήμα κώδικα, χρησιμοποιήσαμε κάποιες επιπλέον συναρτήσεις για να αποφύγουμε επανάληψη του κώδικά μας, όπως φαίνεται παρακάτω.

```
void displayAddressOfItem(struct list *head){
    int value;
    printf("Insert Value of Item to be found:");
    scanf("%d", &value);

    struct list *node;
    node = findItem(head,value);
    if (node == NULL) printf("The item could not be found!");
    else printf("Address of Item: %p", &node);
}

void findAndPrint(struct list *head){
    int value;
    printf("Insert Value of Item to be found:");
    scanf("%d", &value);

    struct list *node;
    node = findItem(head,value);
    if (node == NULL) printf("The item could not be found!");
    else printf("ID of Item: %hd", node->id);
}

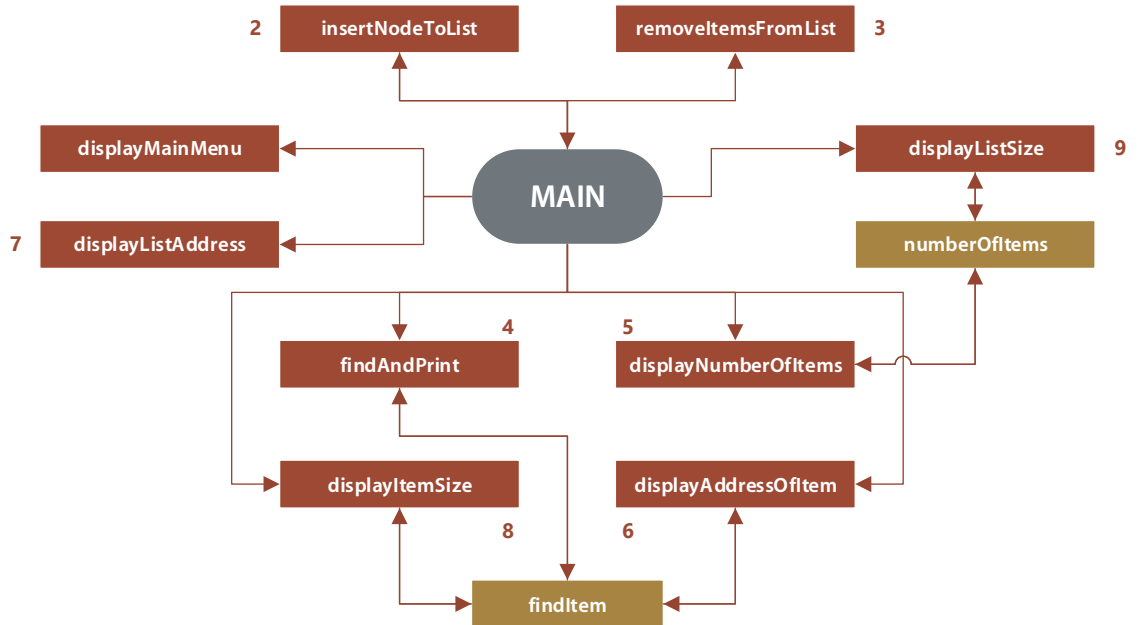
struct list* findItem(struct list *head, int value){
    struct list *currentNode = head;
    if (head == NULL) return NULL;
    if (currentNode->value == value) return currentNode;
    while ((currentNode->next != NULL) && (currentNode->next->value
    if (currentNode->next == NULL) return NULL;
    return currentNode->next;
}
```

Συμπέρασμα

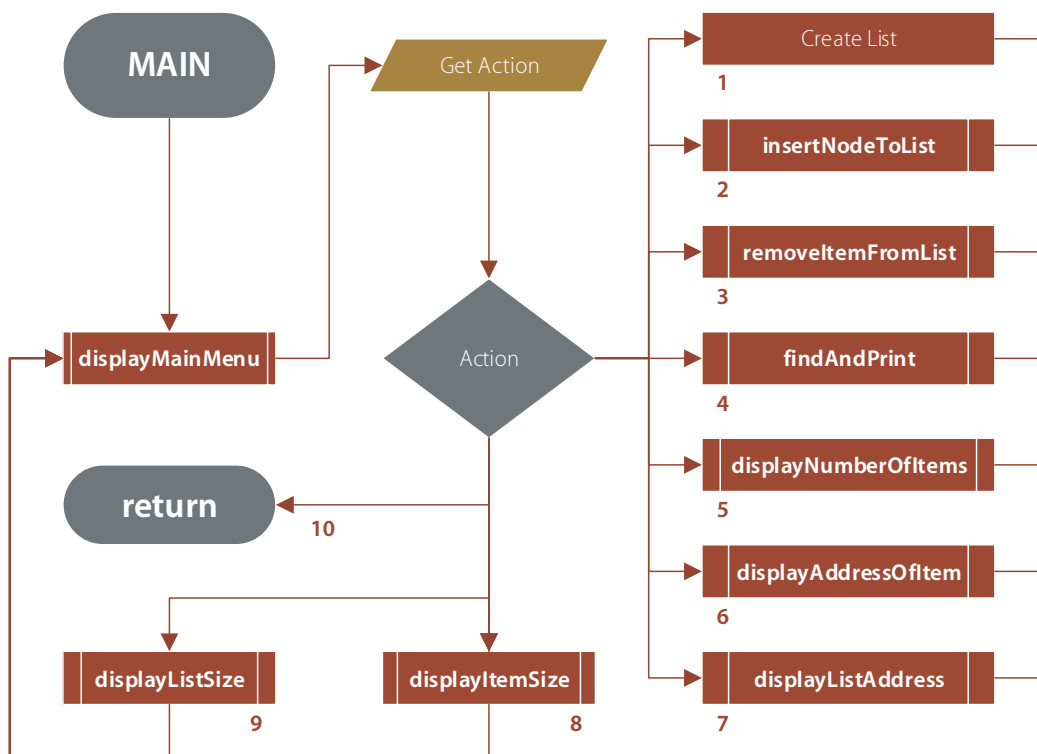
Στο δεύτερο εργαστήριο μάθαμε να δουλεύουμε με τη γλώσσα Clang. Εξοικειωθήκαμε με τα χαρακτηριστικά της και μπορούμε πλέον να αντιμετωπίσουμε πιο πολύπλοκες εφαρμογές.

Παράρτημα – Flowcharts

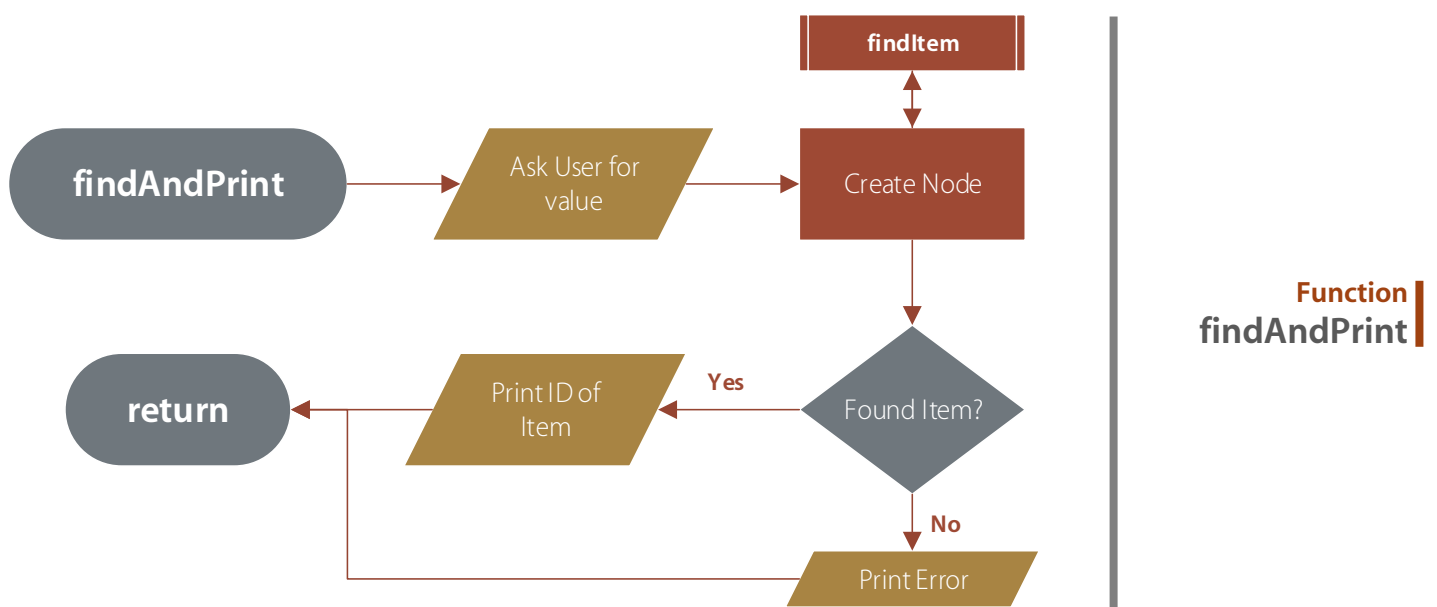
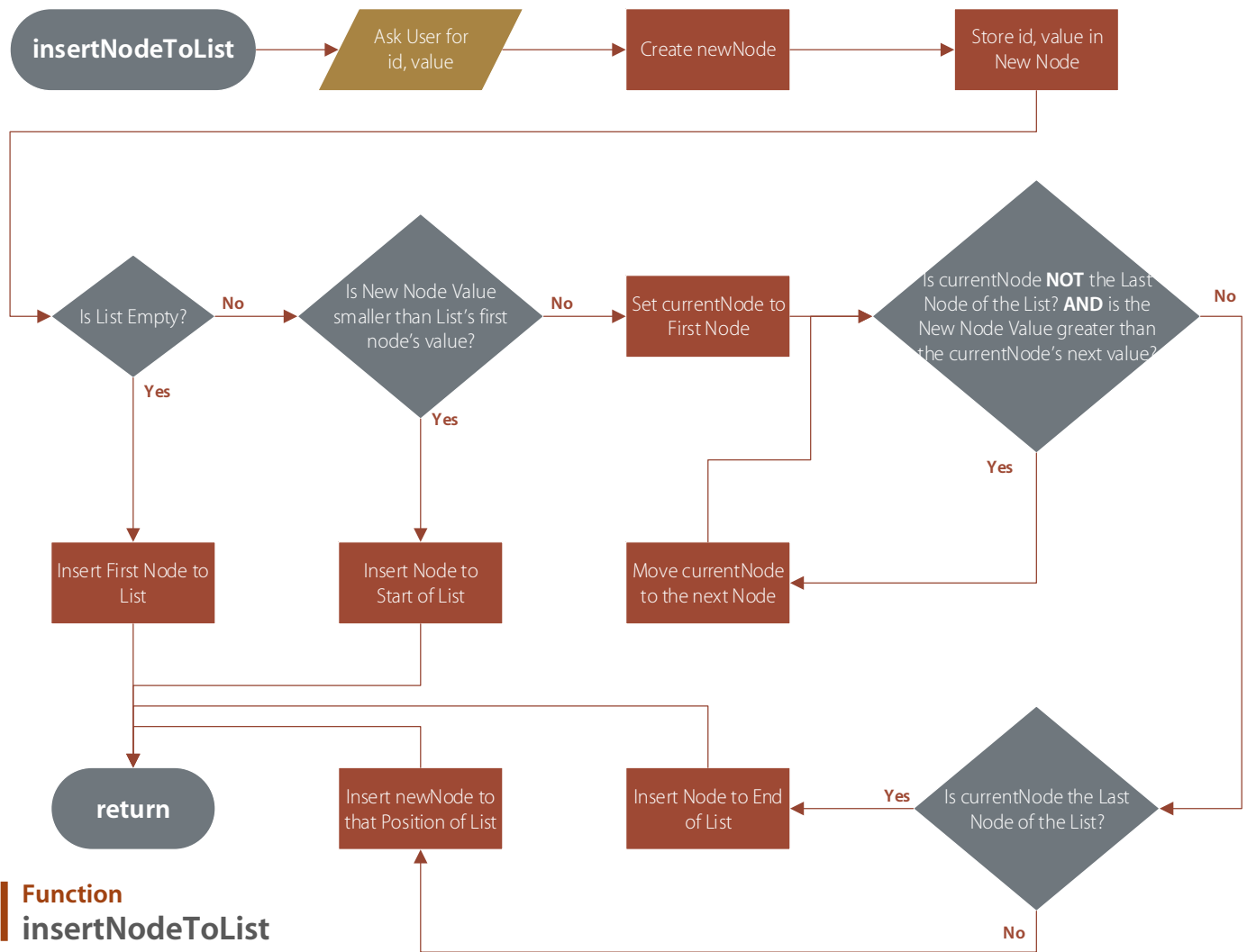
Overview All Functions

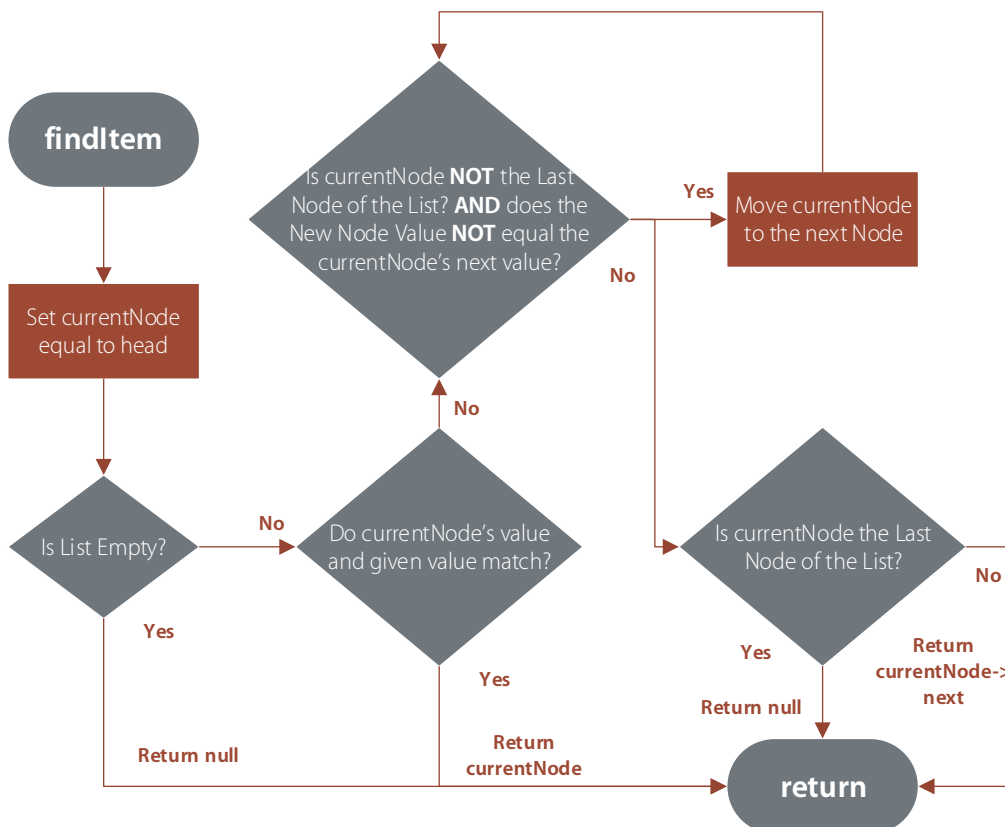
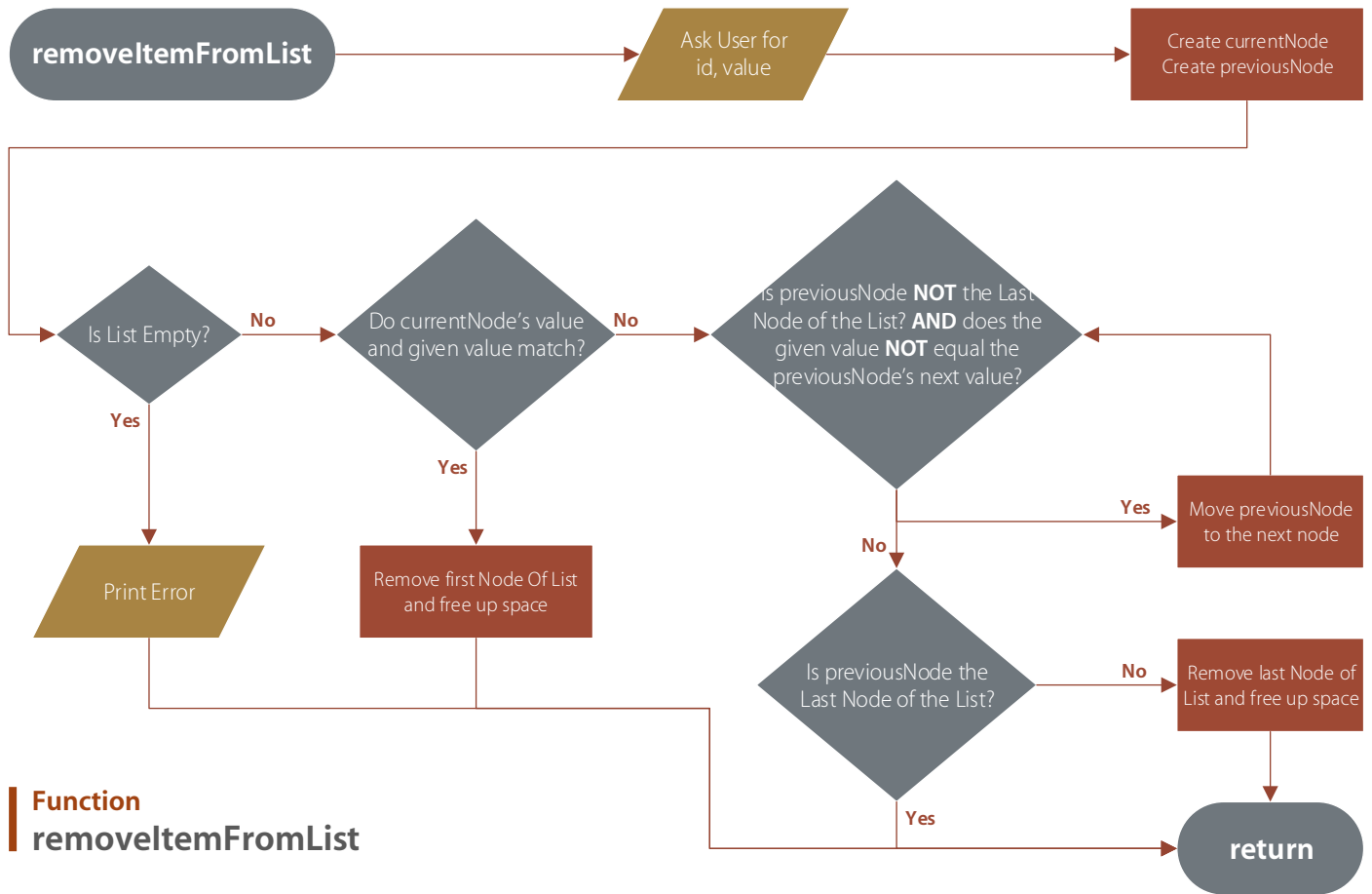


Παρακάτω περιγράφονται μία-μία οι συναρτήσεις του προγράμματός μας.

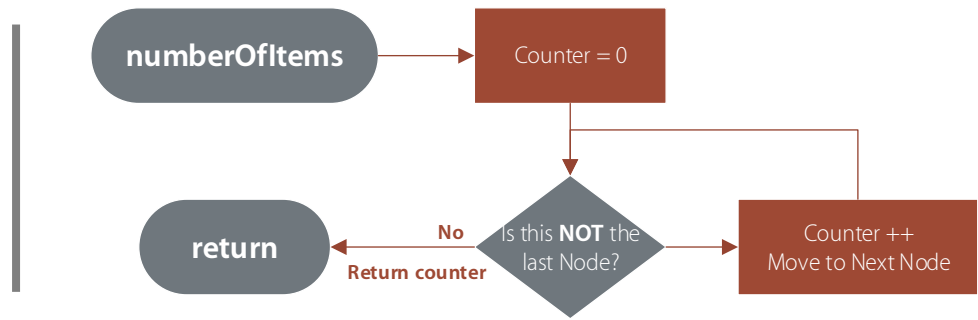


Function Main

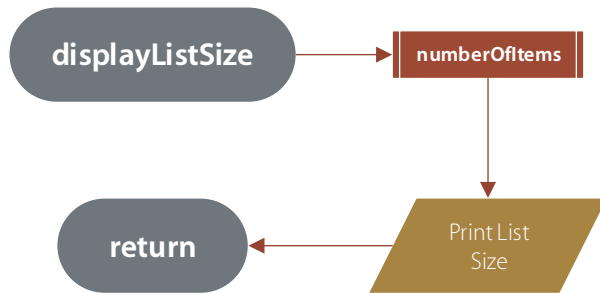




**Function
numberOfItems**

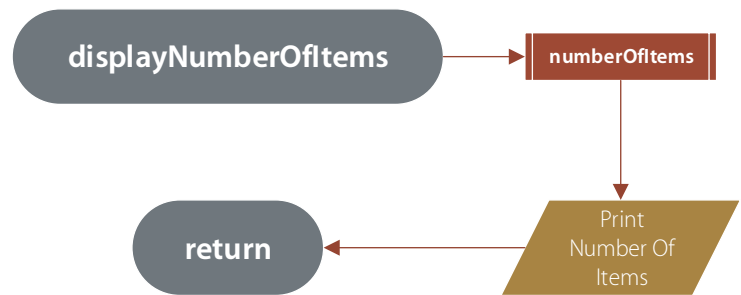


displayListSize



**Function
displayListSize**

**Function
displayNumberOfItems**

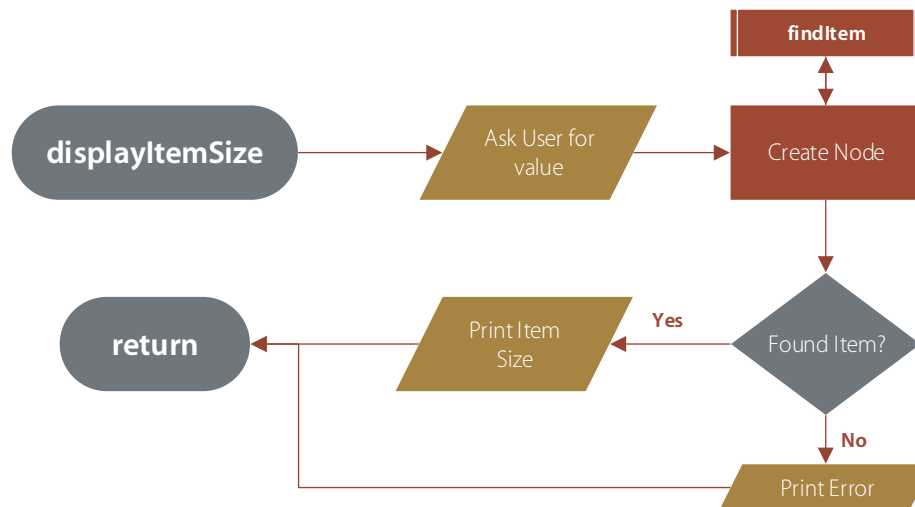
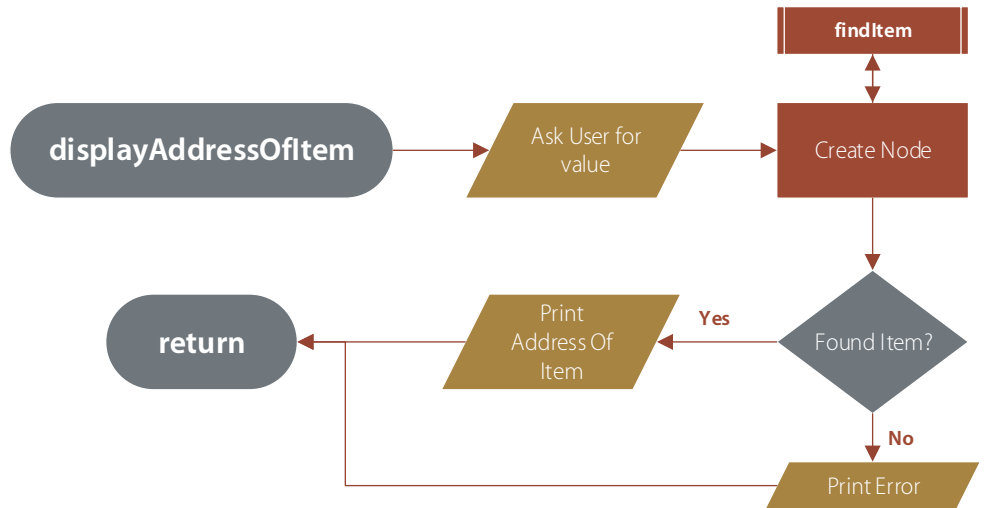


displayListAddress



**Function
displayListAddress**

Function displayAddressOfItem



Function displayItemSize

LEGEND

