**מילון- Dictionary**

* המבנה מבוסס על רשימה מקושרת כל פריט חדש יכנס לסוף (tail) ויינתן לו מפתח(לרוב מספר סידורי) וכל מחיקת פריט ייעשה על פי מפתח.
* מבנה נתונים שמאפשר לייצג שמירת מבנה לפי הכנסה ושליפה של . KEY VALUE

**הפונקציות השימושיות:**

void push(int key, int val)

void pop(int key)

struct Item\* search(int key)

**הקוד עצמו:**

void push(int key, int val)

{

    struct Item\* curr;

    curr = (struct Item\*)malloc(sizeof(struct Item));

    curr->key = key;

    curr->val = val;

    if (head == NULL)

    {

        head = curr;

        tail = curr;

        curr->next = NULL;

        curr->prev = NULL;

    }

    else

    {

        tail->next = curr;

        curr->prev = tail;

        curr->next = NULL;

        tail = curr;

    }

}

struct Item\* search(int key)

{

    struct Item\* ret = NULL;

    struct Item\* curr = head;

    while (curr != NULL)

    {

        if (curr->key == key)

        {

            ret = curr;

            return ret;

        }

        curr = curr->next;

    }

    return ret;

}

void pop(int key)

{

    struct Item\* curr;

    curr = search(key);

    if (curr)

    {

        if (curr == tail && curr != head)

        {

            tail = curr->prev;

            tail->next = NULL;

            free(curr);

        }

        else if (curr == head && curr != tail)

        {

            head = curr->next;

            head->prev = NULL;

            free(curr);

        }

        else if (curr != head && curr != tail && curr != NULL)

        {

            curr->prev->next = curr->next;

            curr->next->prev = curr->prev;

            free(curr);

        }

        else if (curr == head && curr == tail)

        {

            free(curr);

            head = NULL;

            tail = NULL;

        }

    }

    else

    {

        printf("Not found in dictionary!\n");

    }

}