**HW1 QUESTİON 2**

ADT yapabilmek için öncelikle tüm verilerimizi bir paket/yığın haline getirmeliyiz. Bunu C de struct yapısını kullanarak yapabiliriz.

- Public interface ve declaration ları implimentation’u gizlemek için ayrı ayrı dosyalara koymalıyız.

- Her operasyon için ayrı fonksiyonlar kullanmalıyız.

- Type declerationlarını “struct.h” a, public interfaceleri “structfunks.h” a ve implimentasyonları “struct.c” dosyalarına koyabiliriz.

Burada çok çok basit bir şekilde linked list yapısını ADT ye çevirmeye çalıştım. Oncelikle bir structre oluşturudum, bu structure linked list yapısının bir arada kalmasını sağlayacak bir yığın. Bu yığının içerisine operasyon yapacağımız fonksiyonlarımızı koyacağız, “struct.h” da basitçe structureyi oluşturdum. Daha sonra “structfunks.h” ile linked list yapısının public interfacesini oluşturdum. Kullanıcının bilmesi gereken tek şey buradaki interface. Başka hiçbir detayı bilmesine gerek yok. Son olarak ise “struct.c” dosyası ile oluşturmuş olduğum linked list yapısının implimentasyonlarını ekledim. Kullanıcıdan tamamen bu kısım saklanacak bu kısımı bilmesine kullanıcının ihtiyacı yok.

Bu şekilde C de ADT’yi kullanmış olduk.

**struct.h**

struct nodeStruct{

int keyValue;

struct nodeStruct \*next;

}

typedef nodeStruct node;

**structfunks.h**

#include “struct.h”

node \* search(node\* head, int key);

**struct.c**

#include <iostream.h>

#include <structfunks.h>

// Search a node according to its key value

node \* search(node\* head, int key){

//CODES

}