Základy programování (IZP)

10. počítačové cvičenie

Brno University of Technology, Faculty of Information Technology
Božetěchova 1/2. 612 66 Brno - Královo Pole
ilazur@fit.vut.cz

Stiahnite si kostru programu
https://github.com/Jorgen98/IZP/blob/main/cv10/sll_single_file.c

Postupne opravte kód podľa zadania

```
typedef struct {
    int id;
    char *name;
} Object;
```

```
typedef struct item Item;
struct item {
    Object data;
    Item *next;
};
```

```
typedef struct {
    Item *first;
} List;
```

Issue 1 - inicializácia zoznamu

```
/**
 * Inicializace seznamu. Vytvori prazdny seznam.
 */
List list_ctor () {

}
```

Issue 2 - vytvorenie novej položky zoznamu

```
/**
 * Inicializace polozky seznamu. Z objektu vytvori polozku bez
 naslednika.
 */
Item *item_ctor(Object data) {

}
```

Issue 1 - inicializácia zoznamu

```
List list_ctor () {
    List newList;
    newList. first = NULL;
    return newList;
}
```

Issue 2 - vytvorenie novej položky zoznamu

```
Item *item_ctor(Object data) {
    Item *newItem;
    newItem = malloc(sizeof(Item)*1);
    if (newItem != NULL) {
        newItem->data = data;
        newItem->next = NULL;
    }
    return newItem;
}
```

Issue 3 - vloženie novej položky na začiatok

```
/**
 * Vlozi polozku na zacatek seznamu.
 */
void list_insert_first (List * list, Item *i) {

}
```

Issue 4 - je zoznam prázdny?

```
/**
 * Vrati true, pokud je seznam prazdny.
 */
bool list_empty(List * list ) {

}
```

Issue 3 - vloženie novej položky na začiatok

```
void list_insert_first (List * list, Item *i) {
    i->next = list->first;
    list->first = i;
}
```

Issue 4 - je zoznam prázdny?

```
bool list_empty(List * list ) {
    return list->first == NULL;
}
```

Issue 5 - odstránenie prvej položky zoznamu

```
/**
 * Odstrani prvni prvek seznamu, pokud je.
 */
void list_delete_first ( List * list ) {

}
```

Issue 6 - počet položiek zoznamu

```
/**
 * Vrati pocet polozek seznamu.
 */
unsigned list_count(List * list ) {

}
```

Issue 5 - odstránenie prvej položky zoznamu

```
void list_delete_first ( List * list ) {
    if ( list->first != NULL) {
        Item *tmp = list->first->next;
        free( list->first);
         list->first = tmp;
    }
}
```

Issue 6 - počet položiek zoznamu

```
unsigned list_count(List * list ) {
    unsigned cnt = 0;
    Item *tmp = list->first;
    while (tmp != NULL) {
        cnt++;
        tmp = tmp->next;
    }
    return cnt;
}
```

Issue 7 - Nájdenie položky s najmenším ID

```
/**
 * Najde polozku seznamu s nejmensim identifikatorem. Vraci NULL
 , pokud je
 * seznam prazdny.
 */
Item * list_find_minid ( List * list ) {

}
```

Issue 7 - Nájdenie položky s najmenším ID

```
Item * list_find_minid ( List  * list ) {
    if (list_empty( list )) {
        return NULL;
    }

    Item *min_id = list->first;
    for (Item *tmp = list->first->next; tmp != NULL; tmp = tmp->
 next) {
        if (tmp->data.id < min_id->data.id) {
            min_id = tmp;
        }
    }
    return min_id;
}
```

Issue 8 - nájdenie položky s konkrétnym menom

```
/**
 * Najde polozku seznamu s odpovidajicim jmenem objektu. Vraci
 NULL, pokud
 * takova polozka v seznamu neni.
 */
Item *list_find_name(List * list , char *name) {

}
```

Issue 9 - uvoľnenie zoznamu

```
/**
 * Uvolneni seznamu.
 */
void list_dtor ( List * list ) {

}
```

Issue 8 - nájdenie položky s konkrétnym menom

```
Item *list_find_name(List * list , char *name) {
    for (Item *tmp = list->first; tmp != NULL; tmp = tmp->next) {
        if (strcmp(name, tmp->data.name) == 0) {
            return tmp;
        }
    }
    return NULL;
}
```

Issue 9 - uvoľnenie zoznamu

```
void list_dtor ( List * list ) {
    while ( list ->first != NULL) {
        list_delete_first ( list );
    }
}
```

Rozdelenie programu do viacerých súborov

- Podobne, ako využívame knižnice pomocou include, môžeme takéto knižnice sami vytvoriť
- Knižnica sa typicky skladá z .c a .h súborov

```
sll.h – deklaracia funkcii
```

```
sll.c – definicia funkcii
```

```
sll_main.c – vyuziva funkcie z sll.h
```

Preklad a vytvorenie programu

```
Makefile
```

```
$ make
```

Stačí vypracovať jednu variantu

- Vytvorte program podľa seba.