

Вариант 1:

Нека е даден следния шаблон на структура, дефинираща елемент от едносвързан списък:

```
template <typename T>
struct box {T data; box *next;};
```

Задача 1. Да се дефинира подходящо параметризирана функция `fillGaps(L)`, където `L` е указател към първия елемент на линеен едносвързан списък, чиито елементи са цели числа в нарастващ ред. Между всеки два съседни елемента `K` и `L` на списъка функцията да създава и вмъква последователно всички елементи от интервала $[K+1, L-1]$. Т.е. списъкът се превръща в аритметична прогресия с разлика 1.

Пример: Списъкът $1 \rightarrow 3 \rightarrow 4 \rightarrow 8$, ще се преобразува до $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8$.

Задача 2. Да се дефинира функция `void makeFib ([подходящ тип] first)`, където `first` е указател към първия елемент на едносвързан списък `L` с елементи a_0, \dots, a_n . От списъка да се премахнат всички елементи (освен първия и втория), които не са сума на предишните два елемента.

Пример: Списъкът $0, 1, \underline{3}, \underline{5}, 1, 2$ ще се измени до $0, 1, 1, 2$. Подчертани са елементите, които са изтрети.

Задача 3. Да се дефинира функция `void reverseList ([подходящ тип] first, unsigned int n)`. Функцията да обръща реда на първите `n` елемента на едносвързан списък с указател към първия елемент `first`.

Пример: Ако списъкът съдържа елементите $1, 2, 3, 4, 5$ в този ред, при `n=3` списъкът ще се измени до $3, 2, 1, 4, 5$.

Време за работа: 100 минути