

## Задача А1. Как построить минимальный остов?

Ниже приведены три алгоритма  $ALG_1$ ,  $ALG_2$  и  $ALG_3$ , которые из заданного связного неориентированного графа  $G = (V, E)$  выбирают некоторое множество его ребер  $T$ .

```
1  ALG_1(G):  
2      отсортировать ребра графа G  
3      в порядке невозрастания весов;  
4      T = E;  
5      foreach (e ∈ E в порядке невозрастания весов)  
6          if (ребра T - {e} образуют связный граф)  
7              T = T - {e};  
8      return T;
```

```
1  ALG_2(G):  
2      T = ∅;  
3      foreach (e ∈ E, выбранное случайным образом)  
4          if (ребра T ∪ {e} образуют граф без циклов)  
5              T = T ∪ {e};  
6      return T;
```

```
1  ALG_3(G):  
2      T = ∅;  
3      foreach (e ∈ E, выбранное случайным образом)  
4          T = T ∪ {e};  
5          if (в T имеется цикл из ребер  $s \subseteq T$ )  
6              e_max = ребро с максимальным весом  
7                      в цикле s;  
8              T = T - {e_max};  
9      return T;
```

### Система оценки

- 7 баллов Для каждого из трех представленных алгоритмов обоснуйте его наиболее эффективную *по временной сложности* реализацию, в особенности, с точки зрения используемых структур данных и операций над ними. Обоснуйте оценки сложности. Представьте исходный код на языке C++ для каждой из соответствующих реализаций, в которых используемые структуры данных достаточно отразить на уровне интерфейса — приводить полный код используемых структур данных *не нужно*.
- 5 баллов Для каждого из трех представленных алгоритмов определите, формируется ли в множестве ребер  $T$  *минимальное остовное дерево* исходного графа  $G$ . Обоснуйте свой ответ и приведите (контр)примеры.