

## Цифровая олимпиада «Волга-IT'XXI»



### Дисциплина «Автоматизированное тестирование (PHP)»

#### Задание отборочного этапа

Участнику олимпиады в рамках отборочного этапа предлагается протестировать функционал эндпоинта “/triangle/possible” сайта “Possible triangle” <https://possible-triangle.herokuapp.com/> основываясь на предоставленных требованиях.

#### Требования к функционалу

1. Определяет может ли существовать треугольник с такими сторонами;
2. Принимает 3 query параметра: a, b, c;
3. Переданные параметры являются сторонами треугольника;
4. Принимаются только натуральные числа;
5. Программа возвращает ответ в формате json;
6. Если треугольник существует, возвращается ответ:

```
{  
  "isPossible": "true"  
}
```
7. Если треугольник не существует, возвращается ответ:

```
{  
  "isPossible":"false"  
}
```

8. Если переданы не валидные данные, возвращается ответ:

```
{  
  "message": {  
    "error":"Not valid data"  
  }  
}
```

### Для выполнения задания понадобится

1. IDE с поддержкой PHP (например, VS Code);
2. Окружение для запуска PHP кода.

### Порядок действий при выполнении задания

3. Склонировать или скачать репозиторий <https://github.com/barsukov2/test-possible-triangle>;
4. Ознакомиться с примером написания тест кейса `tests/api/GetTesterHiCest.php`  
<https://github.com/barsukov2/test-possible-triangle/blob/master/tests/api/GetTesterHiCest.php>;
5. Продумать тест кейсы для функционала, описанного в параграфе “Требования к функционалу”;
6. Оформить эти кейсы в виде автоматизированного теста на PHP;
7. Убедиться, что он выполняется;
8. Написать отчеты о найденных ошибках.

### Пример запроса

<https://possible-triangle.herokuapp.com/triangle/possible?a=2&b=3&c=4>

## **Что мы хотим получить**

Участник для проверки должен предоставить:

1. Баг репорт в формате PDF/DOC на любом удобном облачном хранилище;
2. Тест кейсы в PHP коде со ссылкой на репозиторий GitHub или Архив проекта;
3. Ссылки сохранить в личном кабинете участника.

## **Критерии оценивания**

Работа будет оцениваться по следующим критериям (максимальное количество баллов 1024):

1. Качество и понятность написания кода – максимально 346 баллов;
2. Качество тестового покрытия (количество проверенных тестовых сценариев) – максимально 286 баллов;
3. Количество найденных ошибок – максимально 196 баллов;
4. Качество составленных отчетов об ошибках – максимально 196 баллов.