**Чек-лист приложения Triangle.**

Для тестирования данного приложения будем применять:

1. Ручное тестирование;
2. Автоматизированное тестирование;

В ходе **ручного тестирования** проверим:

1. Дымовое тестирование:
   1. Запуск приложения с первоначальными параметрами, указанными в методе Main;
2. Тестирование критического пути:
   1. Дополним метод Main треугольниками:
      1. Обычным;
      2. Равнобедренным;
      3. Равносторонним;
      4. Прямоугольным;
      5. a=null, b >0, c>0;
      6. b=null, a >0, c>0;
      7. c=null, a >0, b>0;
      8. a=infinity, b >0, c>0;
      9. b=infinity, a >0, c>0;
      10. c=infinity, a >0, b>0;
      11. (p\*(p-a)\*(p-b)\*(p-c))<0, где p=(a+b+c)/2.

И вызовем методы checkTriangle(), detectTriangle(), getSquare() для указанных треугольников.

В ходе **автоматизированного тестирования** проведем:

1. Модульное тестирование, используя JUnit4:
   1. Тесты для метода checkTriangle(), для этого проверим треугольники:
      1. Любой треугольник, со сторонами >0, <1000;
      2. С одной стороной равной 0 (3 шт.);
      3. С двумя сторонами равными 0 (3 шт.);
      4. Со всеми сторонами, равными 0 (1 шт.);
      5. С одной стороной равной null (3 шт.);
      6. С двумя сторонами равными null (3 шт.);
      7. Со всеми сторонами, равными null (1 шт.);
      8. С одной стороной < 0
      9. С двумя сторонами < 0;
      10. Со всеми сторонами < 0;
      11. С одной стороной равной +бесконечности;
      12. С двумя сторонами равными +бесконечности;
      13. С тремя сторонами равными +бесконечности;
      14. С одной стороной равной -бесконечности;
      15. С двумя сторонами равными -бесконечности;
      16. С тремя сторонами равными -бесконечности;
      17. Со сторонами, при которых выполняются условия:
          1. A+b<=c
          2. A+b>c
          3. A+b = infinity
          4. A+b = -infinity
          5. A+c<=b
          6. A+c>b
          7. A+c=infinity
          8. A+c=-infinity
          9. B+c<=a
          10. B+c>a
          11. b+c = infinity
          12. b+c = -infinity
   2. Тесты для метода detectTriangle(), для этого проверим треугольники:
      1. Со сторонами, при которых выполняются условия:
         1. a\*a + b\*b = c\*c
         2. b\*b + c\*c = a\*a
         3. a\*a + c\*c = b\*b
         4. a=b=c
         5. a=b≠c
         6. a=c≠b
         7. b=c≠a
         8. все вышеуказанные условия не выполняются и a>0, b>0, c>0;
         9. a<0, b>0, c>0
         10. a<0, b<0, c>0;
         11. a<0, b<0, c<0;
         12. a>0, b>0, c<0;
         13. a>0, b<0, c<0;
         14. a>0, b<0, c>0;
         15. a=infinity;
         16. b=infinity;
         17. c=infinity;
         18. a=null;
         19. b=null;
         20. c=null;
   3. Тесты для метода getSquare(), для этого проверим треугольники:
      * 1. a<0, b>0, c>0
        2. a<0, b<0, c>0;
        3. a<0, b<0, c<0;
        4. a>0, b>0, c<0;
        5. a>0, b<0, c<0;
        6. a>0, b<0, c>0;
        7. (a+b+c) = infinity;
        8. (a+b+c) = -infinity;
        9. a=null;
        10. b=null;
        11. c=null;
        12. a+b+c=0;
        13. (p\*(p-a)\*(p-b)\*(p-c))<0, где p=(a+b+c)/2.