1. Tt1

**public** List<Product> findAll() {

List<Product> list = productDAO.findAll();

**for** (**int** i = 0; i < list.size() - 1; i++) {

**for** (**int** j = i + 1; j < list.size(); j++) {

**if** (list.get(i).getProductPrice() > list.get(j).getProductPrice()) {

Collections.*swap*(list, i, j);

}

}

}

**Return li**st;

}

* Đồ thị
* Số đường kiểm thử: 10-8+2=4
* Các đường kiểm thử:

+ 1,2,3,4,5,6,7,8

+1,2,3,4,6,7,8

+ 1,2,3,6,7,8

+ 1,2,7,8

1. Tt2

**public** **boolean** save(Product product) {

**if** (product.getProductId() == **null** || product.getProductId().isEmpty()) {

**throw** **new** Exception("Điền mã sản phẩm");

} **else** **if** (product.getProductId().length() > 6) {

**throw** **new** Exception("Mã sản phẩm tối đa 6 ký tự");

} **else** **if** (product.getProductName() == **null**) {

**throw** **new** Exception("Điền tên sản phẩm");

} **else** **if** (!java.util.regex.Pattern.*matches*(REGEX\_NAME\_PRODUCT, product.getProductName())) {

**throw** **new** Exception("Tên sản phẩm từ 1 - 50 ký tự và chỉ chứa ký tự chữ");

} **else** **if** (product.getProductPrice() <= 0) {

**throw** **new** Exception("Giá sản phẩm phải lớn hơn 0");

} **else** **if** (!java.util.regex.Pattern.*matches*(REGEX\_UNIT\_PRODUCT, product.getProductUnit())) {

**throw** **new** Exception("Đơn vị tính sản phẩm chỉ chứa ký tự chữ");

} **else** **if** (!typeImage.contains(product.getProductImage()

.substring(product.getProductImage().lastIndexOf(".") + 1)

)) {

**throw** **new** Exception("Định dạng file chấp nhận " + typeImage);

}

**if** (productDAO.findById(product.getProductId()) != **null**) {

**throw** **new** Exception("Sản phẩm đã tồn tại");

}

**return** productDAO.save(product);

}

* Đồ thị
* Số đường kiểm thử: 27-19+2=10
* Các đường kiểm thử:

+ 1,2,4,19

+ 1,2,3,4,19

+ 1,2,3,5,6,19

+ 1,2,3,5,7,8,19

+ 1,2,3,5,7,9,10,19

+ 1,2,3,5,7,9,11,12,19

+ 1,2,3,5,7,9,11,13,14,19

+ 1,2,3,5,7,9,11,15,16,19

+ 1,2,3,5,7,9,11,15,16,17,18,19

+ 1,2,3,5,7,9,11,15,16,17,19