Equivalence Class Partitioning (ECP) - All 4 Types Explained Simply

Equivalence Class Partitioning (ECP) is a **black-box testing technique** that divides input values into **valid and invalid classes** to reduce the number of test cases while ensuring full coverage.

There are **four types** of Equivalence Class Testing:

1. **Weak Normal Equivalence Class Testing**
2. **Strong Normal Equivalence Class Testing**
3. **Weak Robust Equivalence Class Testing**
4. **Strong Robust Equivalence Class Testing**

---

1. Weak Normal Equivalence Class Testing

✅ **Single fault assumption** (only one variable changes at a time).
✅ **One test case is selected per equivalence class per variable.**
✅ **Does not include invalid values.**

Example: Age Input (Valid: 18 - 60)

Equivalence Class	Representative Test Case
Valid Age (18 - 60)	30

- Here, we **only** test **one valid value** (e.g., 30).
- **Invalid values (e.g., 17, 61) are not tested.**

👉 **Key Point:** Only valid classes are tested, and only one test case per class is chosen.

---

2. Strong Normal Equivalence Class Testing

✅ **Multiple fault assumption** (tests all possible combinations of valid classes).
✅ **Uses the Cartesian product of all equivalence classes.**
✅ **Does not include invalid values.**

Example: Online Order System

- **Two Inputs:** Payment Type & Shipping Method
- **Valid Equivalence Classes:**
  - **Payment:** (Credit Card, PayPal)
  - **Shipping:** (Standard, Express)

Test Cases (All Valid Combinations)

Test Case	Payment Type	Shipping Method
TC1	Credit Card	Standard
TC2	Credit Card	Express
TC3	PayPal	Standard
TC4	PayPal	Express

👉 **Key Point:** Tests **all possible valid combinations**.

---

3. Weak Robust Equivalence Class Testing

✅ **Single fault assumption** (only one variable changes at a time).
✅ **Includes both valid and invalid values.**
✅ **One test case is selected per class per variable.**

Example: ATM Withdrawal (Valid: $100 - $5000)

Equivalence Class	Representative Test Case
Valid Amount (100 - 5000)	**$1000**
Invalid Amount (<100)	**$50**
Invalid Amount (>5000)	**$6000**

👉 **Key Point:** Both **valid and invalid cases** are tested, but **only one variable at a time**.

---

4. Strong Robust Equivalence Class Testing

✅ **Multiple fault assumption** (tests all possible combinations).
✅ **Includes both valid and invalid values.**
✅ **Tests all possible valid + invalid combinations (Cartesian product).**

Example: Online Order System

- **Payment Types:** (Credit Card ✅, PayPal ✅, Invalid ❌)
- **Shipping Methods:** (Standard ✅, Express ✅, Invalid ❌)

Test Cases (Valid + Invalid Combinations)

Test Case	Payment Type	Shipping Method
TC1	Credit Card ✅	Standard ✅
TC2	Credit Card ✅	Express ✅

Test Case	Payment Type	Shipping Method
TC3	PayPal ✅	Standard ✅
TC4	PayPal ✅	Express ✅
TC5	Invalid ❌	Standard ✅
TC6	Invalid ❌	Express ✅
TC7	Credit Card ✅	Invalid ❌
TC8	PayPal ✅	Invalid ❌
TC9	Invalid ❌	Invalid ❌

👉 **Key Point: Every combination** (valid + invalid) is tested.

Final Summary

Type	Fault Assumption	Includes Invalid Cases?	Test Case Selection
Weak Normal	Single Fault	❌ No	One valid case per class
Strong Normal	Multiple Fault	❌ No	All valid combinations
Weak Robust	Single Fault	✅ Yes	One valid + one invalid per class
Strong Robust	Multiple Fault	✅ Yes	All valid + invalid combinations

---

Conclusion

- **Weak Normal → Tests only valid cases, one per class.**
- **Strong Normal → Tests all valid combinations.**
- **Weak Robust → Adds invalid cases, but tests one at a time.**
- **Strong Robust → Tests every possible combination of valid and invalid cases.**

Would you like a **Python script to generate these test cases automatically**? 🚀