

# Testowanie

## Wprowadzenie

# Plan

- Weryfikacja i walidacja,
- Testowanie,
- Debugowanie,
- Ograniczenia testowania,
- Strategie testowania.

# Weryfikacja

- System technicznie poprawny,
- Produkt budowany prawidłowo.

# Walidacja

- System zgodny ze specyfikacją,
- Budowany jest prawidłowy produkt.

# Weryfikacja i walidacja

- Celem procesu weryfikacji i walidacji jest wykrycie błędów oraz sprawdzenie zgodności produktu z wymaganiami klienta,
- Weryfikacja i walidacja może być wykonywana na różnych etapach projektu,
- Podejście statyczne i dynamiczne.

# Weryfikacja i walidacja

- Celem weryfikacji i walidacji jest uzyskanie określonego poziomu pewności zachowania systemu,
- Nie oznacza to że system jest bezbłędny.

# Testowanie

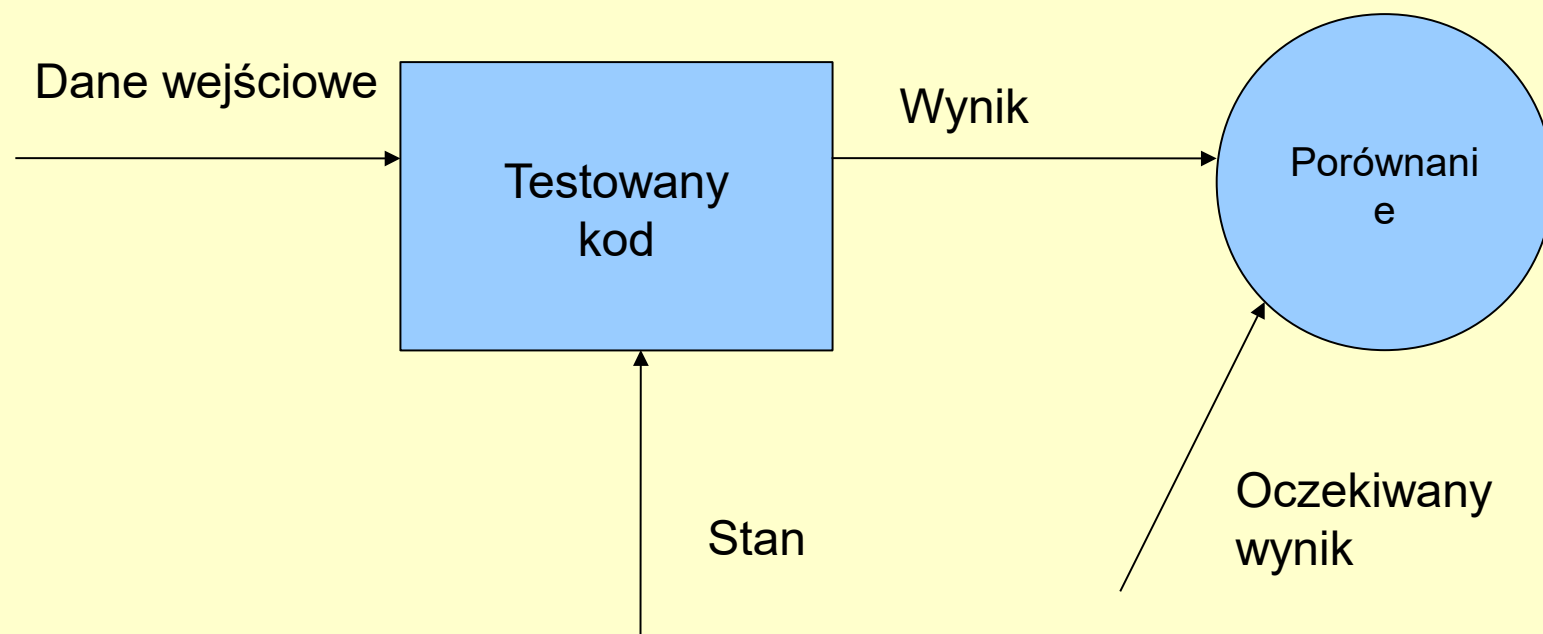
- Testowanie jest weryfikacją dynamiczną,
- Testowanie polega na uruchomieniu programu z danymi wejściowymi oraz porównaniu uzyskanego wyniku z oczekiwanym,
- Testowanie powinno być stosowane wraz z innymi technikami weryfikacji i walidacji.

# Testowanie

- Test jest udany jeśli udało się wykryć błąd,
- Zdolność do wykrywania nowych błędów stanowi o efektywności testu.



# Testowanie



# Testowanie

- Metoda białej skrzynki,
- Metoda czarnej skrzynki.

# Debugowanie

- Celem testowania jest wykrycie błędu,
- Celem debugowania jest lokalizacja źródła błędu i jego usunięcie.

# Debugowanie

- Lokalizowanie błędu często angażuje krokowe wykonanie kodu z punktami zatrzymania (breakpoint),
- Debugowanie z reguły wymaga specjalizowanego programu – debuggera.

# Ograniczenia testowania

- Testowanie może ujawnić obecność błędów, ale nigdy nie zapewni ich braku – Dijkstra,
- Testowanie nie może być jedyną techniką weryfikacji i walidacji,
- Jakość testów.

# Rodzaje testowania

- Testy jednostkowe,
- Testy integracyjne,
- Testy systemowe,
- Testy akceptacyjne,
- Testy interfejsów,
- Testy нефunkcjonalne,
- ...

# Podsumowanie

- Weryfikacja i walidacja,
- Dobry program  $\neq$  bezbłędny program,
- Testowanie to nie wszystko.

# Q&A