

# Bug (Softwarová chyba)

**Bug** je obecné označení pro jakoukoliv chybu v návrhu, implementaci nebo provozu softwaru. Tyto chyby mohou sahát od drobných estetických vad (např. špatně zobrazený text) až po kritické bezpečnostní trhliny, které umožňují vznik [exploitů](#).

Původ termínu je často spojován s legendárním incidentem z roku 1947, kdy operátoři počítače Mark II (včetně Grace Hopperové) našli skutečnou mûru (anglicky „bug“) uvízlou v relé, což způsobilo chybu systému. Termín se však v technickém slangu používal již mnohem dříve, například i Thomasem Edisonem.

---

## Klasifikace chyb

Chyby se dělí podle toho, ve které fázi životního cyklu softwaru vznikají nebo jak se projevují:

### 1. Syntaktické chyby (Syntax Errors)

Vznikají při psaní kódu, kdy programátor poruší pravidla daného programovacího jazyka (např. chybějící středník). Kompilátor takový kód odmítne spustit.

### 2. Logické chyby (Logic Errors)

Program běží, nepadá, ale dává špatné výsledky. Například místo sčítání odečítá. Jsou nebezpečné, protože se hůře odhalují.

### 3. Chyby při běhu (Runtime Errors)

Program se spustí, ale během práce selže (spadne). Častou příčinou je dělení nulou nebo přístup k neexistující části paměti (viz [Buffer Overflow](#)).

---

## Závažnost a priorita

V profesionálním vývoji se každý nahlášený bug hodnotí dvěma parametry:

- **Závažnost (Severity):** Jak velký dopad má chyba na funkčnost? (Např. Kritická = plynulý chod není možný; Nízká = překlep v menu).

- **Priorita (Priority):** Jak rychle musí být chyba opravena? (Často se shoduje se závažností, ale ne vždy – např. oprava loga firmy na hlavní stránce má vysokou prioritu, i když technická závažnost je nízká).

---

## Životní cyklus bugu

Nahlášený bug prochází procesem správy (Bug Tracking):

1. **New (Nový):** Chyba je nahlášena do systému.
2. **Assigned (Přiřazen):** Vývojář začíná na opravě pracovat.
3. **Resolved/Fixed (Vyřešen):** Kód byl opraven.
4. **Verified (Ověřen):** Tester potvrdil, že oprava funguje a nezpůsobila jiné chyby (regresní testování).
5. **Closed (Uzavřen):** Bug je definitivně vyřešen.

---

## Slavné softwarové chyby

- **Y2K Bug:** Obava z přelomu roku 1999/2000, kdy starší systémy ukládaly rok pouze jako dvě cifry („99“), což mohlo vést k nesprávným výpočtům po přechodu na „00“.
- **Ariane 5 (Let 501):** V roce 1996 explodovala raketa krátce po startu kvůli chybě při převodu 64bitového čísla na 16bitové v řídicím softwaru.
- **Therac-25:** Tragická chyba v softwaru ozařovacího přístroje, která kvůli špatné synchronizaci (tzv. race condition) způsobila smrtelné předávkování pacientů radiací.

---

## Prevence a debugging

K minimalizaci chyb slouží:

- **Unit Testing:** Automatizované testy jednotlivých malých částí kódu.
- **Code Review:** Kontrola kódu jiným programátorem.
- **Debugger:** Nástroj, který umožňuje programátorovi zastavit program v libovolném místě a prohlížet obsah paměti.

---

*Související pojmy: Debugging, Exploit, Buffer Overflow, Software Testing, Kompilátor, Zdrojový kód.*

From:

<https://serviceit.cz/> - **IT ENCYKLOPEDIE**

Permanent link:

<https://serviceit.cz/doku.php?id=bug>

Last update: **2025/12/31 19:31**

