



# Moja Odp. Na Apel Twórcy j. C++ Bjarne Stroustrup

2025 Wszelkie Prawa Zastrzeżone przez Jacka Marcina Jaworskiego czyli Energo Kodera Atlanta

autor: [Jacek Marcin Jaworski](#)  
pseudonim: [Energo Koder Atlant](#)  
pomocnicy autora: BRAK  
miejsce: Pruszcz Gd.  
utworzono: 2025-03-24, pon.  
wersja: 2049 z dnia: 2025-03-31  
program składu: Libre Office Writer  
sys. op.: Triskel, Kubuntu

## Spis treści

Wstęp.....	1
1 Podstawa wysokiej jakości prog. to dok. tech. i testy aut.....	1
1.1 Proponowane ograniczenia w stosowaniu SI: tylko optymalizacja i testy aut.....	1
2 Nonsensowna argumentacja w ww art.....	1
3 Podsumowanie.....	2
4 Bibliografia.....	2

## Wstęp

Oto moja odp. na apel autora j. C++ Bjarne Stroustrup z Danii, w którym wzywa społeczność C++ do obrony tego j. prog. O jego apelu dowiedziałem się z art. [tworca-c++-prosi-o-pomoc].

## 1 Podstawa wysokiej jakości prog. to dok. tech. i testy aut.

Moim zdaniem nie ma "cudownego sposobu" na eliminację błędów w kodach prog. pisanych przez ludzi. Można za to wskazać sposoby ich wykrywania i poprawiania.

Najważniejszymi narzędziami programisty pomocnymi w prog. wysokiej jakości kodu są: specyfikacja

techniczna i testy automatyczne. Moim zdaniem wszelkie problemy z atakami typu "buffer-overflow" wynikają albo z nieprecyzyjnej specyfikacji technicznej, albo ze zbyt słabych testów aut. I nie ma to związku z żadnym j. prog. Po prostu niedbałe prog. bez ich dopracowania i tak będą niedbałe i dziurawe. Jedyne co może się okazać konieczne to wydłużenie procesu projektowania i testowania oprogramowania. Tak by nadać mu wysoką jakość. Na pewno należy dopracowywać wszelkie formaty i protokoły stosowane w sieci Internet. Być może do testów aut. należało by używać sztucznej inteligencji, która metodami siłowymi testowała by 1) wart. typowe, 2) skrajne, ale dopuszczalne i 3) wart. które nie są dopuszczalne w specyfikacji technicznej.

## 1.1 Proponowane ograniczenia w stosowaniu SI: tylko optymalizacja i testy aut.

Ani do proj. prot. sieciowych, ani do proj. formatów danych ani do kodowania nie używał bym sztucznej inteligencji z tego powodu, że to sprawia zagrożenie utraty przez ludzi kompetencji w tych dziedzinach.

Sztucznej inteligencji używałbym wył. do optymalizacji (wyszukiwania optymalnych kształtów przedmiotów i optymalnych param. pracy) i do testów aut.

## 2 Nonsensowna argumentacja w ww art.

cytat [tworca-c++-prosi-o-pomoc]:

"Presja ze strony regulatorów

Nie bez znaczenia są także zmiany regulacyjne. Amerykańska CISA opublikowała w październiku ubiegłego roku raport zalecający, by do 1 stycznia 2026 r. producenci mieli plan eliminacji luk pamięciowych lub przeszli na języki bezpieczne dla pamięci. Stroustrup uznał to za "realne zagrożenie" dla C++.

Rząd USA proponuje następujące języki:

Rust  
Go  
C#  
Java  
Swift  
JavaScript  
Ruby"

Pow argumentacja jest nonsensowna z tego powodu, że na tej liście tylko Rust i Go są j. prog. a reszta to j. skryptowe. Obie grupy j. są nieporównywalne i mają inne zastosowania.

## 3 Podsumowanie

Moja końcowa myśl jest taka, że wygląda na to że trafne są moje wcześniejsze przewidywania, które opublikowałem w [f-35-zasady-kodowania] i że faktycznie rząd SZAP/USONA chce wycofać z rynku cywilnego j. C++ i że chcą zrobić z niego kol. "czarny proj.". Bo na przekór pow. propagandzie SZAP/USONA uruchamia kol. proj. militarne oparte o prog. kodowane w j. C++ (np. patrz art. w [ntw-2024-11] i wzmiankę o przepisaniu z j. ADA na j. C++ oprogramowania nowych wer. rakiet samosterujących dalekiego zasięgu).

Znamy to z przeszłości: wcześniej tak było np. z sys. op. Plan9, który był stworzony przez twórców sys. op. Unix. Sys. op. Plan9 był genialny i przełomowy, jednak porzucony. Co było kol. proj. twórców sys. op. UNIX? Pozostaje tajemnicą bez odp. Jednak ja to wiem: następca sys. op. Plan9 należał do "czarnych proj." czyli tajnych i to na zawsze.

Innym przykładem wycofywania rozw. z cywilnego rynku są dżojstiki używane przez domowych graczy komputerowych. Teraz dżojstiki są używane wyłącznie w wojsku. Natomiast w domach do gier komputerowych używa się nieergonomicznych padów.

Problemy z niedbałymi prog. kodowanymi w j. C i C++ mogą stać się pretekstem do utajnienia kol. wer. tych j. prog.

## 4 Bibliografia

### Bibliografia

tworca-c++-prosi-o-pomoc: Paweł Czajkowski, Twórca C++ prosi o pomoc. Przyszłość języka zagrożona przez rząd USA i Rust, 2025,  
[https://ithardware.pl/aktualnosci/tworca\\_c\\_jezyk\\_zagrozony\\_usa\\_rust-39456.html](https://ithardware.pl/aktualnosci/tworca_c_jezyk_zagrozony_usa_rust-39456.html)  
f-35-zasady-kodowania: Jacek Marcin Jaworski, F-35 - Zasady Kodowania - komentarz, 2025,  
<https://energokod.pl/komentarze-do-ksiazek/2025-01-19%20F-35%20-%20Zasady%20Kodownia%20-%20komentarz.pdf>  
ntw-2024-11: Michał Gajzler, JASSM niejedno ma imię, czyli AGM-158B-2, B-3, D oraz LRASM i JASSM-XR, 2024