

C++. Podróż Po Języku - komentarz



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

autor:	Jacek Marcin Jaworski
pseudonim:	Energo Koder Atlant
utworzono:	2022-11-01
wersja: 240 z dnia:	2026-06-28
Miejsce:	Pruszcz Gd., Polska
system:	Linux (distro: Kubuntu)
program:	LibreOffice
źródło:	energokod.gda.pl

Ten dok. w wer. PDF jest podpisany cyfrowo wolnym prog. GNU gpg dostępnym bezpłatnie ze s. www.gnupg.org/download. Instrukcja w j. pol. jak się posługiwać prog. GNU gpg w sys. Linuks z rodz. Debian/Ubuntu znajduje się w całkowicie bezpłatnym dok. PDF [Konf. i Zabezp. Sys. Op. z Rodz. Ubuntu](#) w roz. "Podpisywanie Dok. PDF".

Spis treści

Wstęp.....	1
Skróty.....	1
Autor Ks. Pt. „C++. Podróż Po Języku”.....	1
Adresat Ks. Pt. „C++. Podróż Po Języku”.....	2
Forma Ks. Pt. „C++. Podróż Po Języku”.....	2
Treść Ks. Pt. „C++. Podróż Po Języku”.....	2
Podsumowanie.....	3
Rewelacje.....	3
Zalety.....	3
wadY.....	3
partactwA.....	3
Wnioski.....	3
Licencja.....	3
Bibliografia.....	3

Wstęp

W tym dok. skomentuję ks. pt. „C++. Podróż Po Języku Dla Zaawansowanych” aut. p. Bjarne Stro-

ustrup, wyd. III z w 2023r. przez wyd. Helion z Gliwic, ISBN 978-83-8322-525-8, w tłumaczeniu Łukasza Piwko.

Tytuł org. „A Tour of C++ (C++ In Depth Series) 3rd Edition” wyd. Peason Education, Inc, jako Addison Wesley Professional w 2023r.

W dalszym tekście będę stosował spolszczoną nazwę org. czyli „C++. Podróż Po Języku”. Bo dopisek polskiego wydawcy „Dla Zaawansowanych” to kiepski żart.

Skróty

arg.	argument
aut.	autor
dok.	dokument
el.	element
f.	funkcja
j.	język
Jed.	jednostka
jw	jak wyżej
kl.	klasa
kol.	kolejność
ks.	książka
odp.	odpowiedź
org.	oryginał
p.	punkt
pam.	pamięć
pt.	pod tytułem
pyt.	pytanie
roz.	rozdział
s.	strona
SI	sztuczna inteligencja
sys. op.	system operacyjny
wer.	wersja
wyd.	wydanie, wydawnictwo

Skrótów 4literowych i dłuższych nie tłumaczę, bo uważam je za oczywiste.

Autor Ks. Pt. „C++. Podróż Po Języku”

Autorem ks. jest sam autor j. C++ czyli słynny Bjarne Stroustrup. Od końca lat 1970tych do chwili obecnej bierze udział w pracach nad tym j. prog.

Aktualnie Bjarne jest pod silną presją ze strony obozu rządzącego w SZAP/USONA, który domaga się skuteczniejszych rozw. (na poziomie j. prog.) przeciwko niektórym atakom krakerskim bazujących na przepełnianiu buforów w pam. operacyjnej. Jako szczególnie podatne na te ataki rząd SZAP/USONA wskazuje j. prog. C i C++ (czytaj np. [tworca-c++-prosi-o-pomoc]).

Adresat Ks. Pt. „C++. Podróż Po Języku”

Adresatem ks. są programiści znający j. C++, ale w starszych wer. i chcący poznać najnowsze udogodnienia, które pojawiły się w od spec. C++2011 do spec. C++2020.

Forma Ks. Pt. „C++. Podróż Po Języku”

Ks. jest wydania na śnieżnobiałym papierze (nieznanej kl.), w miękkich okładkach.

Treść Ks. Pt. „C++. Podróż Po Języku”

Biorąc ks. do ręki od razu wiemy, że jest to ks. kierowana do ludzi, a nie dla moli książkowych ani dla jakiejś SI. Wynika to z tego, że ks. ma 312s. co oznacza, że autor się streszcza i przestrzega retoryki i poetyki, czyli ekonomii językowej. Na tle innych ks. o C++ to OGROMNY PLUS!

Po otwarciu ks. mamy kol. pozytywne wrażenie: przejrzysty spis treści. Widać, że roz. mają od 10 do 20s. Co oznacza, że każdego dnia można przeczytać jeden roz. Jest to zdrowe, bo kończąc roz. wiemy jak on się rozp. Natomiast gdyby lektura roz. trwała np. tydzień to nie było by już to takie oczywiste.

Po spisie treści widzimy, że autor postanowił omówić praktycznie cały j. C++. Dla mnie osobiście b. ciekawym jest pyt.:

Czy ks. w takiej postaci była by wystarczającym podręcznikiem dla osoby chcącej się nauczyć prog. w j. C++ od podstaw?

Gdy się zastanowiłem nad tym pyt., to zrozumiałem, że świetnym uzupełnieniem tej ks. był by zbiór zadań do samodzielnej realizacji z wykorzystaniem określonych technik pokazanych w ks. Taki zbiór zadań można też samemu sobie wymyśleć. I mam zamiar wykonać taki proj. w przyszłości.

Natomiast już obecnie wytypowałem sobie najważniejsze nowości podane w ks. i, w oparciu o nie, stworzyłem jed. nauki w moim prog. do nauki (działa on na zasadzie SuperMemo).

Po lekturze ks. zmieniłem swoje nastawienie do nowości w C++. Wcześniej wydawało mi się, że „nowoczesny kod C++” przestaje nawet wyglądać jak C i że zmienia się w jakiegoś niezrozumiałego potwora. Jednak teraz widzę, że autor C++ chce rozwijać ten j. w poszanowaniu największych zalet składni j. C. Tak więc obecnie rozszerzenia j. wprowadzone od C++2011 do C++2020 postrzegam jako użyteczne innowacje, a nie jako zaprzeczenie wcześniejszych silnych p. j. C i C++.

Wielką zaletą są porady na końcu każdego roz.

Należy też powiedzieć, że oprócz nowości w składni C++ aut. omawia też nowości w bibl. STL (w przestrzeni nazw std::).

B. dobrym pomysłem było dodanie roz. 19. Historia i Zgodność, bo dzięki temu aut. dokumentuje historię rozwoju j. C++.

W ks. dostrzegłem też pewne drobne wady:

1. Na s. 15. f.

```
int main(){}
```

nic nie zwraca. Jest to coś co zostało z j. C, ale moim zdaniem jest to wbrew logice. Tak samo jak np. użycie nie zadeklarowanej zmiennej czy f. też uważane jest za sprzeczne z logiką.

2. Na s. 60. kod:

```
if (s<0)
```

tłumacz określił jako „nieujemny”.

3. Na s. 70. operator+() i operator-() modyfikują swoje arg. co wg mnie jest b. złą praktyką w prog.

4. Aut. omawia rozw. jakie nie były dostępne w czasie pisania ks.: moduł std, oraz f. finally(). Aby ich użyć trzeba samemu je stworzyć, a to wymaga dodatkowej pracy (zwy-

kle oznacza to nieprzewidziane kłopoty i konieczność dodatkowego testowania).

5. cyt. s. 146.: *Ponadto, gdy program używa wielu łańcuchów o różnej długości, może dojść do fragmentacji pamięci.* Jest to rodzaj manipulacji, bo takie rzeczy jak koszt de-fragmentacji pamięci ma znaczenie jedynie w sys. czasu rzeczywistego, a popularne sys. op. nie są takimi sys. (ani Linuks, ani Windows, ani MacOS, ani AmigaOS nie są sys. czasu rzeczywistego).

6. Na s. 210 jest taki kod:

```
void user(forward_range auto& r)
{
    int count = 0;
    for (int x : r)
        if (x % 2) {
            cout << x << ' ';
            if (++count == 3) return;
        }
}
```

Dla mnie taki kod łamie „zasadę linii produkcyjnej” (czyli zasadę etapowego przetwarzania), bo jest w nim pomieszane filtrowanie el.

`(x % 2) != 0`

oraz wypisywanie tych el. na konsolę (jako wynik działania prog.). Ten kod jest pomieszany i nieczytelny. Dlatego nie jest to dobra praktyka inżynierska ani edukacyjna. Więcej na ten temat piszę w mojej monografii [arch-prog-nieup] w roz. „Zasada linii produkcyjnej”.

Podsumowanie

Rewelacje

1. Opis nowości j. C++ przez aut. samego twórcy tego j. prog.;
2. B. dobry pomysł na ks.: Krótko i na temat.

Zalety

1. B. dobry papier.

wady

1. Drobne niedoskonałości w treści – jw.

partactwa

1. BRAK

Wnioski

Jak zwykle w przypadku zagranicznych ks. problemem jest, to że b. szybko się je zapomina. To zmusza do użycia spec. technik pamięciowych w celu zapamiętania najważniejszych nowości. Ja w tym celu używam swojego prog. TurboRetty. Jednak to za mało. Należy zdobyć zestaw ćwiczeń z C++2020 lub samemu je sobie wymyśleć i przetrenować najważniejsze nowości w praktyce. Bo to **praktyka czyni mistrzem!**

Licencja

Jest to licencja dotycząca tego dokumentu. Pliki wskazywane przez linki mogą być publikowane na innych licencjach. Zasady licencji:

1. **Zezwolenie na kopiowanie** Zezwala się na niekomercyjne kopiowanie tego dokumentu;
2. **Zezwolenie na udostępnianie** Ten dokument można udostępniać (jednak bezpłatnie);
3. **Zabronione modyfikowanie** Tego dokumentu nie można modyfikować ani skracać ani dodawać czegokolwiek.
4. **Ograniczenia licencji nie dotyczą autora.**

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
arch-prog-nieup: Jacek Marcin Jaworski, Arch. Prog. Nieuprzywilejowanych, 2026,
<https://energokod.gda.pl/monografie/Arch.%20Prog.%20Nieuprzywilejowanych.pdf>