

## Jak podglądać pliki nagłówkowe języka 'C'?

Pliki nagłówkowe Dev C++ w moim komputerze (u Ciebie będzie podobnie jeżeli nie wręcz tak samo):

C:\Program Files (x86)\Dev-Cpp\MinGW64\x86\_64-w64-mingw32\include

Możesz wybrany przez siebie plik nagłówkowy natychmiast otworzyć w środowisku Dev C++.

**Tak dostaję treść dowolnego pliku nagłówkowego, np. `stdio.h` ze środowiska Dev C++ :**

Wywołuję 'Wiersz polecenia' czyli 'command prompt' instrukcją **cmd** tam, gdzie jest ikona lupy ze słowem 'Wyszukaj' :



Jakąkolwiek sekwencję zobaczysz, nie przejmuj się nią i od razu napisz **cd/** przechodząc na sam dół ścieżki:

```
C:\Users\Artur>cd/  
C:\>cd Program Files (x86)\Dev-Cpp\MinGW64\x86_64-w64-mingw32\include  
C:\Program Files (x86)\Dev-Cpp\MinGW64\x86_64-w64-mingw32\include>stdio.h
```

I od razu otwiera mi się wybrany plik nagłówkowy (tu: **stdio.h**) w edytorze Dev C++. Możesz go zapisać w swoim folderze aby mieć do jego treści bezpośredni dostęp.

**Uwaga:** U Ciebie ścieżka dostępu do plików nagłówkowych może się trochę różnić ale sekwencja zapisu:

...MinGW64\x86\_64-w64-mingw32\include\Twój plik nagłówkowy.h

wyduje mi się w każdej wersji załadunku Dev C++ utrzymana.

## Dlaczego akurat język 'C'?

Każdy względnie nowy język programowania jest tworzony pod określonym kątem jego zastosowania.

Trudno mi sobie wyobrazić bardziej wszechstronny ze wszystkich niż język 'C'.

Widać to zarówno po aplikacjach tworzonych w oparciu o ten język jak i możliwych jego zastosowaniach.

Nie tak dawno temu ograniczenie pamięci i mocy obliczeniowej stawiane przez powszechnie dostępne komputery uniemożliwiało prowadzenie operacji, nawet tych akademickich i wymagało komputerów specjalnego przeznaczenia takich jak mainframe, z których 'Odra 1305' miała moc obliczeniową około 270 tys. operacji na sekundę. Obecne laptopy wykonują grubo ponad miliard operacji zmiennoprzecinkowych na sekundę. To już wystarcza aby każdy z programistów mógł kontrolować procesy w reaktorach chemicznych lub obliczyć ciąg silników rakietowych...

Do tego nie jest ani Java, ani HTML, ani Python. To potrafią tylko język 'C' i jego klony.

## Źródła materiałów do opracowania tej strony internetowej:

- Turbo C++, Version 3.0 User's Guide; Borland International, Inc; 1992; stron 772; ISBN (brak)
- Brian W. Kernighan, Dennis M. Ritchie, The C Programming Language, Second Edition; Prentice Hall Software Series; 1988; stron 272; ISBN 0-13-110362-8 (pbk.), 0-13-110370-9
- Jan Bielecki, Encyklopedia Języka C dla IBM PC, tom 1 - Opis Języka, Wydawnictwa Komunikacji i Łączności; 1989; stron 480; ISBN 83-206-0919-4
- Jan Bielecki, Encyklopedia Języka C dla IBM PC, tom 2 - Biblioteki, Wydawnictwa Komunikacji i Łączności; 1989; stron 320; ISBN 83-206-0919-4
- Jan Bielecki, Turbo C z grafiką dla IBM PC; Wydawnictwo Naukowo-Techniczne; 1990; stron 404; ISBN 83-204-1101-7
- Alex Ragen, Leksykon Języka C; Wydawnictwa Naukowo-Techniczne; 1990; stron 160; ISBN 83 204-1223-4

- D. S. Malik, C++ Programming: Program Design Including Data Structures, Course Technology, Thomson Learning; 2002; stron 1404; ISBN 0-619-03569-2
- Robert Lafore, The Waite Group's C Programming Using Turbo C++, Second Edition; SAMS Publishing; 1993; stron 778; ISBN 0-672-30399-X
- Robert Lafore, The Waite Group Object-Oriented Programming in C++, Second Edition; Waite Group Press; 1995; stron 863; ISBN1-878739-73-5
- Tom Swan, Mastering Borland C++ 4.5, Second Edition; SAMS Premier; 1995; stron 1461; ISBN 0 672-30546-1
- Mitchell Waite and Stephen Prata, The Waite Group's New C Primer Plus, Second Edition; SAMS Publishing; 1993; stron 736; ISBN 0-672-30319-1
- Namir Clement Shammass, Teach Yourself Visual C++ 1.5 in 21 Days, Revised Edition; SAMS Publishing; 1994; stron 923; ISBN 0-672-30489-9
- Peter Aitken & Bradley Jones, Teach Yourself C in 21 Days, Bestseller Edition; SAMS Publishing; 1994; stron 720; ISBN 0-672-30448-1
- Narain Gehani, C: An Advanced Introduction (ANSI C Edition); Computer Science Press; 1988; stron 265; ISBN 0-7167-8196-4
- Steve Schustack, Variations in C, Second Edition; Microsoft Press; 1989; stron 427; ISBN 1-55615 239-6
- Steven Holzner, C++ Black Book; Coriolis; 2001; stron 799; ISBN 1-57610-777-9
- Paul J. Perry, Do It Yourself, Turbo C++, First Edition; SAMS Publishing; 1992; stron 571; ISBN 0 672-30107-5
- Jesse Liberty, Teach Yourself C++ in 24 Hours, First Edition; SAMS Publishing; 1997; stron 450; ISBN 0-672-31067-8
- Stephen Prata, Mitchell Waite Signature Series: C++ Primer Plus, Third Edition; White Group Press; 1998; stron 1019; ISBN 1-57169-131-6
- Peter D. Hipson, Advanced C, First Edition; SAMS Publishing; 1992; stron 777; ISBN 0-672-30168 7

Polecam ze względu na wyjątkową łatwość czytania, możliwość weryfikacji kodów i dużą wartość poznawczą:

1. Dokument Stanford University próbujący podsumować wszystkie podstawowe aspekty języka C. Tylko 45 stron; nie przeszkadza, że po angielsku bo to inne spojrzenie na wszystkie opisywane przeze mnie tematy.  
<http://cslibrary.stanford.edu/101/EssentialC.pdf>
2. C code-reference (Programming Reference/Libraries).  
Chociaż nie wszystkie tematy są tu wykończone, znajdziesz tu sporo krótkich, przejrzystych kodów.  
<https://code-reference.com/c>
3. Richard Johnsonbaugh and Martin Kalin: Application Programming in ANSI C - też krótkie kody  
<https://condor.depaul.edu/~mkalin/ed3/>
4. Programowanie w Dev C++.  
38 stron. To raczej zeszyt do ćwiczeń z języka C, ale po polsku.  
[https://kry5refy.github.io/cpp\\_dev\\_inst/cpp\\_dev\\_instrukcja\\_Cz\\_1\\_2.pdf](https://kry5refy.github.io/cpp_dev_inst/cpp_dev_instrukcja_Cz_1_2.pdf)
5. Programowanie w języku C dla początkujących oraz średnio zaawansowanych programistów.  
272 strony. Podręcznik języka C, tak przyjemnie napisany, aż się chce to wszystko wkuć na pamięć.  
<https://www.arturpyszczuk.pl/files/c/pwc.pdf>
6. Dodatkowo, bo dla C++, klasyka polskiej literatury informatycznej, pełnej polotu i fantazji: "Od Zera do Gier Kodera", 558 stron, od przyziemnych spraw do wysokich lotów.  
<http://www.cs.put.poznan.pl/arybarczyk/Kurs%20C++.pdf>