

# Reforma Modelu Biznesowego Wolnego Oprogramowania

2023-2024 Wszelkie Prawa Zastrzeżone przez Jacka Marcina Jaworskiego czyli Energo Kodera Atlanta

autor:	<a href="#">Jacek Marcin Jaworski</a>
pseudonim:	Energo Koder Atlant
pomocnicy autora:	BRAK
miejsce:	Pruszcz Gd., Polska
utworzono:	2023-01-17, wto.
wersja: 384 z dnia:	2025-11-11
program składu:	Libre Office Writer
sys. op.:	Triskel, Kubuntu
źródło:	<a href="http://energokod.gda.pl">energokod.gda.pl</a>

Ten dok. w wer. PDF jest podpisany cyfrowo wolnym prog. GNU gpg dostępnym bezpłatnie ze [s. www.gnupg.org/download](http://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 całości bezpłatnym dok. PDF [Zabezp. Sys.Debian-Ubuntu](#) w roz. 13 Podpisywanie dok. PDF.

## Spis treści

Wprowadzenie.....	1
O Autorze.....	1
Skróty.....	2
1 Obecny Model Biznesowy.....	2
2 Powody Reformy Modelu Biznesowego.....	2
Eskalacja Kosztów.....	2
Konieczność Zarabiania Na Siebie.....	2
Obecny Model Biznesowy Po Prostu Nie Działa.....	3
Walor Edukacyjny To Fikcja.....	3
Materiały Edukacyjne Są Bardzo Słabe.....	3
Brak Dystrybucji FSF/GNU To Hipokryzja, Podobnie Jest z Linux Foundation.....	3
Brak Możliwości Wykupienia Wsparcia.....	3
Postępująca Izolacja FSF/GNU.....	3
3 Rodzaje Dostarczanych Prog. i Usług.....	3
Rodzaje Wolnego Oprogramowania.....	3
Rodzaje Usług Świadczonej Przez Dostawcę Wolnego Oprogramowania:.....	4
Oczekiwanie Wobec Dostawcy Wolnego Oprogramowania. .4 Referencyjny Sys. Op. Powinien Mieć.....	4
4 Grupy Uż. Wolnego Oprogramowania.....	5
5 Nowy Model Biznesowy.....	5
Ulgi.....	5
6 Licencja.....	5
7 Bibliografia.....	6

## Wprowadzenie

Po ponad 35 latach funkcjonowania FSF sytuacja Wolnego Oprogramowania nie napawa optymizmem. Na uczelniach o nim nic się nie mówi: ani o licencji, ani o ideologii

(za to uczy się sztuczek za pomocą konsoli). W branżowych serwisach informacyjnych można przeczytać jedynie o nowościach dotyczących programu głównego Linuks oraz kompilatora GNU g++ - i to w sytuacji istnienia tysięcy projektów i setek dystrybucji Wolnego Oprogramowania!!! Po podejmowanych przez FSF projektach widać świetne pomysły na stworzenie nowych wolnych systemów i nowych wolnych programów zapewniających prywatność i bezpieczeństwo - widać też, że projekty FSF są głodzone z braku środków na ich rozwój.

Po ponad 35 latach możemy powiedzieć, że nie udało się Ideologii Wolnego Oprogramowania zaistnieć społecznie nawet w polskim środowisku szkolnym ani akademickim. Natomiast to co jest uderzające, to bezwstydne pasożytowanie na udostępnionym publicznie dorobku Ruchu Wolnego Oprogramowania. Pasożytowanie to polega na braku jakiegokolwiek aktu wdzięczności za wykonaną pracę ani za dostarczone rozwiązania: w firmach w jakich pracowałem nikt nawet nie wspomniał by dać choćby parę dolców FSF. Nic się w nich nie mówiło o Ideologii ani o licencji. Nikt nie mówi nawet dobrego słowa o tych co tworzą Wolne Oprogramowanie - a wręcz przeciwnie wygląda to tak, jak by ci co udostępniają kod byli frajerami! Dowodem potwierdzającym zorganizowane pasożytnictwo na Ruchu Wolnego Oprogramowania jest utrzymywanie go w izolacji od reszty społeczeństwa i kreowanie wszystkich programistów (nie tylko od W.O.) na "zidiociały czarowników komputerów".

Widać wyraźnie, że problem nie leży w Ideologii Wolnego Oprogramowania ani w umiejętnościach jego programistów - bo oni podejmują nawet najtrudniejsze wyzwania. Widać, że problem leży w modelu biznesowym, bo on powinien potwierdzać siłę Ideologii i torować drogę do jej promowania w kolejnych pokoleniach i do uzyskania wsparcia do dalszego działania od wszystkich zainteresowanych przedsiębiorstw.

## O Autorze

Programuję komputery od lut. 1997r. Od 1999r. Mam tytuł Technika Elektronika spec. Systemy Komputerowe. Zaliczyłem 3 lata studiów informy na PG (jednak dyplomu inż. nie zrobiłem). Legalną maturę zdałem w 2016r. Jednak po 2005r. wprowadzono w całej Polsce zmiany w prog. studiów zaocznych (wymuszono na wszystkich kier. 7sem. z j. ang.) powoduje to, że nie jestem w stanie ich zaliczyć.

W latach 1999-2007 byłem programistą aplikacji w C++ dla sys. Windows.

W latach 2017-2022 byłem programistą aplikacji w C++ na sys. Linuks i Android.

**Jeśli chodzi o mój kierunek rozwoju, to chcę być jak najlepszym inżynierem i architektem sys. i to nie tylko komputerowych.**

Moje zainteresowania zawodowe to przede wszystkim:

- \*Studia nad architekturą wzorcowych sys. komp.\*
- \*Studia nad nowymi algorytmami (jednak nie SI)\*
- \*Studia nad bezpieczeństwem współczesnych sys. komp.\*
- \*Studia nad prywatnością użytkowników współczesnych sys. komp.\*
- \*Podnoszenie jakości, efektywności i bezpieczeństwa w pracy przez realizację racjonalizatorskich projektów
- \*Zdobycie biegłości w pracy z sys. op. Plan9\*
- \*Programowanie w językach: Asembler, C, C++ i D.

Wystrzegam się jak mogę „produktów” wielkich korporacji, które z niejawnych (zło jest niejawne) powodów zwykle wynajdują patologiczne wynalazki<sup>1</sup>.

Jestem poszukiwaczem i kolekcjonerem "dobrych zasad życiowych" i "dobrych zasad inżynierskich". Dzięki tym związyłem, hasłowym zasadom często widzę sens podejmowania większego wysiłku by uzyskać obiektywnie dobry efekt zamiast stosowania półśrodków.

**Zdrowe zasady pozwalają zostawiać za sobą działające rozwiązania zamiast partactw.**

Jestem praktykującym zwolennikiem filozofii dobra i moralności, czyli Totalizmu. Z filozofią Totalizmu, w wyd. prof. Jana Pająka z NZ, po raz pierwszy zetknąłem się w 2002r. i od razu b. się nią zafascynowałem i tak jest do d. dzisiejszego. Od 2020r. rozwijam jej popr. wer. w postaci [Ideologia Geniuszy-Mocarzy]. Mój Totalizm różni się od Totalizmu prof. Jana Pająka z NZ tym, że ja uważam że Totalizm składa się z dobra i z moralności, a nie z samej moralności. Wynika to z faktu, że sama moralność bez dobra prowadzi do filozofii moralności i zła czyli w istocie do filozofii nazistowskiej.

## Skróty

el.	element
kat.	katalog
NZ	Nowa Zelandia
odp.	odpowiedź.
org.	oryginał
pow.	powyżej

<sup>1</sup> Jak wiecie język C i sys. **Uniks** stworzyło 2 ludzi: **Dennis Ritchie** i **Ken Thompson**. Dlatego jestem na 100% pewien, że gdyby ich projekt był prowadzony z rozmachem w stylu wielkich korporacji, to: a) Język C nigdy by nie powstał, b) **Uniks** nigdy by nie powstał, c) powstały by potwory podobne do **Ada**, **Jawa**, **C#**, **Windows** i **Android**.

**Jak zauważył prof. Jan Pająk z NZ: W organizacjach pasożytniczych wszystkie złe pomysły pojawiają się od góry, a wszystkie postępowe koncepcje powstają oddolnie.**

pyt.	pytanie
ref. sys. opr.	referencyjny system operacyjny
s.	strona
sys. opr.	system operacyjny
uż.	użytkownik
wer.	wersja

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

## 1 Obecny Model Biznesowy

Obecny model biznesowy jest związany z licencją na której publikuje się W.O.: GPL i LGPL. Sprowadza się to do upubliczniania "za darmo" prog. i całych sys. op., jednak bez dokumentacji proj.

Takie postępowanie motywuje się chęcią redukcji barier w dostępie do opr. oraz zapewnienia możliwości jego dowolnego dostosowania do własnych potrzeb. Dodatkowo mówi się, że udostępnianie kodów źródłowych ma walor edukacyjny.

## 2 Powody Reformy Modelu Biznesowego

### Eskalacja Kosztów

W.O. już dawno stało się całą gałęzią gospodarki a nie tylko plikami źródłowymi publikowanymi przez zwolenników W.O. Nakładów finansowych wymaga:

1. Integracja opr. w jeden, spójny i stabilny system operacyjny;
2. Dystrybucja instalacyjnych obrazów systemu i pakietów;
3. Wprowadzanie łątek bezpieczeństwa i nowych wer. programów.
4. Dostawcy W.O. często prowadzą działalność marketingową, budują społeczność w okół dystrybucji oraz zapewniają darmowe poradnictwo.

**To jest potężna eskalacja kosztów! Tego wszystkiego nie da się zrobić w garażu!**

## Konieczność Zarabiania Na Siebie

Okazuje się, że podstawowym wskaźnikiem poziomu czyjejś moralności jest po prostu odp. na pyt. "Czy zarabiasz na siebie?" i "Czy nikogo nie krzywdzisz?".

## Obecny Model Biznesowy Po Prostu Nie Działa

Wiele (większość?) firm w branży oraz wielu ludzi korzysta z W.O. bez jakiegokolwiek wkładu własnego - nie wspierają go nawet moralnie.

To jest czyste pasożytnictwo. A pasożytnictwo jest jedynym źródłem zła na Ziemi i we Wszechświecie. Dobro wynika z wielu źródeł i można podążać przez życie wieloma dobrymi ścieżkami - natomiast zło wynika jedynie z pasożytnictwa i pasożytnictwo jest jedyną drogą zła (więcej w [totalizm-8p-monografia]). Tolerowanie na sobie pasożytnictwa jest grzechem poświęcenia. Poświęcenie jest niemoralne (więcej w [totalizm-8p-monografia]). Zaś poświęcenie twórców W.O. prowadzi do ich izolacji i wyzysku przez pasożyty.

**Pasożyty finalnie zawsze dążą do izolacji ofiary (w celu wyłączności na żerowanie) a następnie do śmierci żywiciela!**

Gdy się nad tym pasożytnictwem zastanawiam, to dochodzę do wniosku, że jedynie osoby uczące się i szkoły są poza tym procederem. Wynika to z tego, że osoby uczące nie mają jeszcze kompetencji by pracować, ale starają się je uzyskać. Natomiast szkoły te kompetencje dają (przynajmniej w teorii). Tak więc uczniowie i szkoły dążą do moralnego celu jakim jest podjęcie pracy przez absolwentów po skończeniu szkoły lub uczelni. Dlatego uczniom, szkołom, studentom i uczelniom należy się specjalne wsparcie. Jednak zepsucie społeczne (demoralizacja) jest tak daleko posunięta, że nawet w odniesieniu do osób uczących się i szkół trzeba wprowadzić formalną kontrolę ich podejścia do W.O. Gdy się tego zaniedba efekt ideologiczny będzie żaden!

## Walor Edukacyjny To Fikcja

Wynika ona z faktu braku dostępnej dok. proj. ogromu kodów źródłowych - nawet w "prostych" programach konsolowych. Braku audytu proj. który by odp. na pyt. "Czy prog. jest prawidłowo zaprojektowany, zakodowany i udokumentowany?"

W.O powinno być przede wszystkim projektowane pod kątem łatwości zrozumienia i czystości architektury, algorytmów i kodu. Wydajność sys. i prog. może być mniejsza gdy optymalizacja degenerowała by architekturę kodu.

Można by też promować poprawę jakości kodu programów organizując konkursy na najdoskonalszy proj. i prototyp danego programu.

Można by też uznać, że w skład systemu operacyjnego powinny wchodzić wył. prog. kodowane w j. kompilowanych, a skrypty powinny być tworzone samodzielnie przez użytkowników.

## Materiały Edukacyjne Są Bardzo Słabe

Rzadko przetłumaczone, często stare i nie pełne, np. [sec-debian-man] ostatnia zmiana jest z kwi. 2017 czyli z przed 7,5r.?!!!!.

## Brak Dystrybucji FSF/GNU To Hipokryzja, Podobnie Jest z Linux Foundation

Ref. sys. op. jest najbardziej podstawową sprawą. Tym bardziej jest to niezrozumiałe, że FSF obecnie na swoich serwerach udostępnia w pełni wolne dystrybucje, np. [trisquel - s. WWW].

Co ciekawe Linux Foundation reklamuj się jako ponad 900 otwartych proj., ale referencyjnego sys. op. brak.

## Brak Możliwości Wykupienia Wsparcia

Brak możliwości wykupienia wsparcia uż. Brak możliwości wykupienia wsparcia dla programisty zainteresowanym użyciem bibl., albo adaptacją lub rozwojem danego prog.

## Postępująca Izolacja FSF/GNU

Cytat z biuletynu "Fall 2022" str. 7: "W.O. podlega ciągłym atakom co jest ostrzeżeniem, które warto powtarzać."<sup>3</sup>. To wynika z tego, że pasożyt izoluje ofiarę, czyli W.O. To dlatego nawet nowości FSF nie są opisywane w formie artykułów. Np. o FSF słyszy się tylko w kontekście skandali w okół Pana Ryszarda oraz nowych wydań GNU gcc/g++ - a przecież podejmowanych problemów i proj. jest dużo więcej!

## 3 Rodzaje Dostarczanych Prog. i Usług

### Rodzaje Wolnego Oprogramowania

Poziomy opr. pod względem architektury:

1. Prog. gł. i jego sterowniki;
2. Demony: lokalne i sieciowe;

<sup>2</sup> Dziś mamy 2024-12-17.

<sup>3</sup> W org. po ang.: "Free software is under constant attack, which is warning worth repeating."

3. Prog. nieuprzywilejowane: Polecenia konsoli i programy interaktywne.

## Rodzaje Usług Świadczonech Przez Dostawcę Wolnego Oprogramowania:

1. Dystrybucja obrazów instalacyjnych;
2. Repozytoria pakietów;
3. Aktualizacje;
4. Wsparcie uż.;
5. Wsparcie programisty;
6. Proj. na indywidualne zamówienie.

## Oczekiwania Wobec Dostawcy Wolnego Oprogramowania

1. Regularny audyt kodu i działania wszystkich dostarczanych prog. wg. jasnych, sprawdzalnych i powtarzalnych kryteriów:

2. np. czytelność architektury, czytelność kodu, bezpieczeństwo, prywatność, wydajność, jakość inst. uż., jakość dok. proj., łatwość zrozumienia proj. i jego podatność na adaptację i rozszerzanie;

**Łatwość zrozumienia proj. powinna mówić jakie wykształcenie i jakie doświadczenie jest zalecane w celu jego pełnego zrozumienia.**

3. Na podst. audytu: certyfikowanie wszystkich dostarczanych prog.;

4. Załączanie certyfikatów do paczek wszystkich dostarczanych prog.;

5. Publikowanie ref. sys. op. Nie powinien on mieć dziesiątek tys. pakietów. Powinien zawierać minimalną konieczną ilość pakietów ale nie mniej<sup>4</sup>. Powinny być dostarczane po JEDNYM najlepszym prog. w danej kat. Aby stwierdzić który prog. jest najlepszy należy przeprowadzić audyty proj. oraz testy użyteczności.

## Referencyjny Sys. Op. Powinien Mieć

1. Komplet dokumentacji:

"Jakie były najważniejsze wydarzenia w historii ziemskiej komputeryzacji?";

"Jakie są najważniejsze koncepcje zastosowań współczesnych sys. komputerowych?";

"Jakie są najważniejsze elementy współczesnych sys. komputerowych?";

"Jakie są najważniejsze koncepcje sieci IP/TCP?";

"Jakie są najważniejsze el. sys. op.?",

"Jakie są najważniejsze zasady bezpieczeństwa i prywatności w pracy z komp. w domu i w firmie?";

"Jak dostosować swój sys. op.?",

"Jak zabezpieczyć własny sys. op.?".

2. Komplet narzędzi do łatwego przetwarzania tekstów (zarówno w linii komend jak i w trybie graficznym);

3. Komplet narzędzi do prac administracyjnych (w linii komend);

4. Komplet klientów sieciowych (zarówno w linii komend jak i w trybie graficznym);

5. Komplet narzędzi do tworzenia opr. (zarówno w linii komend jak i w trybie graficznym);

6. Komplet narzędzi do przetwarzania najpopularniejszych plików multimedialnych (zarówno w linii komend jak i w trybie graficznym);

7. Komplet narzędzi do pracy biurowej w domu i w małej firmie (jednak bez prog. księgowych i magazynowych);

8. Komplet podst. demonów sieciowych (DNS, NFS, Samba, WWW, FTP, poczta el., Torrent, TOR).

9. Możliwość instalacji opr. komercyjnego. Jednak nie może to być tak jak teraz z paczkami Snap i Flatpack gdzie deklarowany zakres dostępu (manifest) do systemu i do danych w katalogu uż. jest tajny i narzucony przez dostawcę - jest to nie dopuszczalne!!!

10. Prog. zewnętrznych dostawców muszą być podatne na pracę w piaskownicy takiej jak Ogniste Więzienie<sup>5</sup>, czyli pod pełną kontrolą uż. Dlatego obecnie na sys. rodz. Debian/Ubuntu w grę wchodzi tylko paczki deb i AppImage.

11. Aktualizacje, czyli dostarczanie poprawek błędów i łątek bezpieczeństwa przez czas wsparcia na zasadach komercyjnych:

11.1. Aktualizacje muszą gwarantować niezmiennosc prog. i niezmiennosc API i ABI bibliotek. Zmiany mogą dotyczyć wyłącznie naprawy błędów i łątek bezpieczeństwa. Wynika to z faktu, że Ubuntu robi różne numery uż.: Nawet w wer. LTS wesoło aktualizuje do najnowszych wer. Jajko Linuksa, Ognistego Lista, Grzmocącego Ptaka. W repo sys. Kubuntu 20.04 latem 2022 (czyli ponad 2 lata od wydania) nawet zlinkowali ze wszystkimi prog. KDE bibliotekę KWallet! Co ciekawe jest ona linkowana a nie używana - sam sprawdziłem w kodzie prog. Krusader.

11.2. W Przypadku sys. op., jego sterowników i demonów aktualizacje powinny wychodzić przez min. 12 lat. Bo 10 lat to oczekiwany czas działania komp (tak uczono mnie w gdańskim Technikum Łączności w latach 1994-99);

11.3. W Przypadku prog. nieuprzywilejowanych aktualizacje powinny wychodzić przez min. 5 lat. 5 lat to standardowy czas życia produktu w branży el.

12. Udostępnianie prog. podst. razem z kodem, inst. uż. i dok. proj. Na zasadzie gratisu;

13. Udostępnianie prog. w wer. pełnej, razem z kodem, inst. uż. i dok. proj., na licencji komercyjnej i na zasadach komercyjnych;

14. Świadczenie wsparcia uż. na zasadach komercyjnych;

15. Świadczenie wsparcia prog. na zasadach komercyjnych;

16. Realizowanie proj. na indywidualne zamówienie na zasadach komercyjnych.

4 Zgodnie z zasadą sys. Unix, po ang.: KISS.

5 po ang.: firejail

## 4 Grupy Uż. Wolnego Oprogramowania

1. Uż. nieuczący się;
2. Uż. uczący się;
3. Uż. biznesowi;
4. Szkoły;
5. Budżetówka z wyjątkiem szkół.

## 5 Nowy Model Biznesowy

1. Dostawca W.O. powinien być organizacją komercyjną, która powinna mieć normalny biznes plan. Organizacja powinna mieć co najmniej 4 zespoły inżynierskie: 1) zespół szybkiego reagowania do łatania krytycznych błędów w prog., 2) zespół systematycznie poprawiający niekrytyczne błędy w prog., 3) zespół systematycznie badający jakość i bezp. prog., 4) zespół rozwijający prog. Organizacja powinna mieć możliwość akumulacji kapitału w celu zgromadzenia rezerwy, zapewnienia finansowania pow. zespołów proj., oraz w celu rozszerzania działalności na nowe obszary oraz do realizacji wszystkich pozostałych zadań i celów statutowych.

2. Wszyscy uż. mają prawo do bezpłatnego pobrania plików z obrazami instalek z ref. sys. op. i do bezpłatnej instalacji.

3. Wszyscy uż. mają prawo do bezpłatnego pobrania kodów źródłowych prog. podst. razem z inst. uż. oraz z dok. proj.

4. Pozostałe usługi są płatne z wyjątkami dla osób uczących się i szkół.

5. Odpłatność należy się za:

5.1. Aktualizację wszystkich pakietów z opr.;

5.2. Pełne wer. (o ile takie są) demonów i prog. nieuprzywilejowanych. Odpłatność dotyczy każdego prog. z osobna.

**Pełne wer. prog. muszą być udostępniane na komercyjnej licencji w celu ochrony prawnej. To samo dotyczy ich kodów źródłowych i dok. proj.**

Ważne jest by uż. zawsze mógł wpływać na kod prog. w sposób bezpośredni, czyli samodzielnie lub chociaż w sposób pośredni, czyli przez płatne zlecenia.

Można określić warunki upubliczniania kodu źródłowego i dok. proj. pełnych wer. prog., np.: osiągnięcie określonego poziomu zysków ze sprzedaży lub kwotę wykupu przez społeczność.

5.3. Wsparcie uż.;

5.4. Wsparcie programisty;

5.5. Realizacji proj. na indywidualne zamówienie.

### Ulgi

1. Uż. nieuczący się: nie mają żadnych ulg (dotyczy to osób prywatnych, firm i budżetówki);
2. Uż. uczący to tacy którzy zapiszą się na formalny semestralny kurs mają prawo do 6 mieś. ulgi.

3. Warunek otrzymania ulgi: Przyznanie przez lokalny MOPS bonu na używanie W.O. danego dostawcy przez 9mieś. na podst. kontroli dok. potwierdzającego status ucznia lub studenta i niski poziom dochodu w rodzinie. Kontrola przez MOPS była by obejściem problemu ujawniania danych osobowych uczniów przez szkoły. Jest to też przerzucenie na MOPS spr. kto jest uczniem a kto nim nie jest.

Uczący się mają nast. ulgi:

3.1. Zwolnienie z opłat za dostęp do repo w sieci Internet;

3.2. Zwolnienie z opłat za aktualizację opr.;

3.3. Przysługuje im bezpłatnie pakiet prog. zgodny z ich kier. nauki w wer. pełnej - chodzi o to by wspierać uczących się w ich wybranej specjalizacji;

3.4. Przysługuje im bezpłatnie jeden dodatkowy pakiet prog. w wer. pełnej z dowolnej innej dziedziny - chodzi o to by wspierać rozwój innych zainteresowań;

3.4.1. Uczący może zmienić dodatkowy pakiet prog. w wer. pełnej w ciągu 1. mieś. obowiązywania ulgi.

4. Szkoły to jednostki realizujące naukę w trybie min. semestralnym. Ulga należy się szkole pod nast. warunkiem:

4.1. Wprowadzenia przedmiotu na wszystkich nauczanych kier na temat Ideologii W.O., podst. koncepcji bezpieczeństwa i prywatności w trakcie używania sys. komp. w globalnej sieci komp. oraz podst. koncepcji programistycznych.

4.2. Przedmiot ten powinien być realizowany w wymiarze min. jedno semestralnego kursu w trybie dziennym i wieczorowym: min. 2 godz. na tydzień lub w trybie zaocznym: min. 2 godz. co 2 tygodnie;

4.3. Program nauczania na tym przedmiocie powinien zostać wstępnie zaproponowany przez dostawcę W.O., a następnie przetworzony w danej szkole tak by dostosować go do lokalnych warunków i poziomu kształcenia, finalnie ten program powinien zostać zatwierdzony przez dostawcę W.O.

Spełnienie pow. warunków powinno nastąpić przed rozpoczęciem roku szkolnego lub roku akademickiego (w zależności od tego czy to szkoła czy uczelnia).

Szkołom, które spełnią pow. wymagania, należy się nast. ulgi:

4.4. Zwolnienie z opłat za dostęp do repo w sieci Internet;

4.5. Zwolnienie z opłat za aktualizację oprogramowania;

4.6. Zwolnienie z opłat za korzystanie z pakietów prog. w pełnych wer. zgodnych z nauczonymi kier.

## 6 Licencja

Jest to licencja dotycząca tego dokumentu. Pliki wskazane 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 cokolwiek.

4. Ograniczenia licencji nie dotyczą autora.

## 7 Bibliografia

### Bibliografia

Ideologia Geniuszy-Mocarzy: Jacek Marcin Jaworski,

Ideologia Geniuszy-Mocarzy, 2025,

<https://energokod.pl/monografie/Ideologia>

%20Geniuszy-Mocarzy.pdf

totalizm-8p-monografia: prof. Jan Pająk, Totalizm, 2004

sec-debian-man: praca zbiorowa, Securing Debian  
Manual, 2017

trisquel - s. WWW: , Trisquel GNU/Linux RSS, 2024,

<https://trisquel.info>