

# XX OI, zawody III stopnia — Krótkie wprowadzenie do Linuksa

---

## Wstęp

Drodzy zawodnicy!

Ten krótki podręcznik powstał z myślą o tych spośród Was, którzy nie mają specjalnego doświadczenia w pracy pod Linuksem. Mamy nadzieję, że okaże się pomocny.

## Pisanie programów

Do pisania swojego kodu możesz wykorzystać jedno z dostępnych środowisk programistycznych takich jak Geany czy Code::Blocks. Aby uruchomić dowolne z nich, wybierz z menu Programy podmenu Programowanie. Na początek polecamy użycie Geany.

Do tworzenia swoich rozwiązań możesz też wykorzystać proste edytory tekstowe, takie jak Gedit czy Kate. Znajdziesz je w menu Programy, w podmenu Akcesoria.

Podczas pisania programów w C/C++ częstym błędem jest włączanie do kodu bibliotek dostępnych tylko w systemie Windows, takich jak `conio.h`, lub umieszczanie na końcu programu komendy `system("pause");`. Jej użycie jest niepotrzebne, ponieważ po zakończeniu programu wynik jego działania nadal będzie widoczny w terminalu.

## Terminal

W środowisku Linux ważnym narzędziem jest terminal. Aby go uruchomić, z menu Programy wybierz Akcesoria, a następnie Terminal.

Po uruchomieniu będziesz w swoim folderze domowym, czyli `/home/zawodnik`.

- Aby przejść do podfolderu, wpisz:

```
$ cd nazwa_podfolderu
```

- Aby wrócić do folderu nadrzędnego, wpisz:

```
$ cd ..
```

- W każdym momencie możesz sprawdzić, w jakim folderze się znajdujesz:

```
$ pwd
```

## Kompilacja

Aby skompilować swój program, w terminalu wejdź do folderu, w którym zapisałeś swoje rozwiązanie, i uruchom jedną z poniższych komend:

- dla języka C:

```
$ gcc -O2 -static -lm -o nazwapliku nazwapliku.c
```

- dla języka C++:

```
$ g++ -O2 -static -lm -o nazwapliku nazwapliku.cpp
```

# XX OI, zawody III stopnia — Krótkie wprowadzenie do Linuksa

---

- dla języka Pascal:

```
$ gcc386 -O2 -XS -Xt nazwapliku.pas
```

W wyniku jej działania w folderze, w którym się znajdujesz, powstanie plik o nazwie `nazwapliku`.

Jeśli kompilujesz program napisany w języku C lub C++, warto rozważyć dodanie flagi kompilatora `-Wall`. Jej użycie spowoduje bardziej szczegółowe wypisywanie ostrzeżeń podczas kompilacji, co pozwala odnaleźć wiele błędów.

Jeśli swój kod piszesz w środowisku programistycznym takim jak Geany czy Code::Blocks, to do kompilacji możesz użyć wbudowanej w niego opcji (np. w Geany wystarczy wybrać z menu Zbuduj opcję Zbuduj).

## Uruchamianie

Aby uruchomić swoje rozwiązanie, uruchom terminal oraz wejdź do folderu, w którym zapisałeś swój kod. Jeśli tego wcześniej nie zrobiłeś, skompiluj rozwiązanie. Wpisz:

- `$ ./nazwaprogramu`  
aby uruchomić swój program
- `$ ./nazwaprogramu < plik_testowy.in`  
aby uruchomić swój program oraz przekierować zawartość pliku `plik_testowy.in` na standardowe wejście
- `$ ./nazwaprogramu < plik_testowy.in > wynik.out`  
aby uruchomić swój program, przekierować zawartość pliku `plik_testowy.in` na standardowe wejście oraz wynik działania programu zapisać do pliku `wynik.out`
- `$ time ./nazwaprogramu < plik_testowy.in > wynik.out`  
aby uruchomić swój program, przekierować zawartość pliku `plik_testowy.in` na standardowe wejście, wynik działania programu zapisać do pliku `wynik.out` oraz dodatkowo wyświetlić czas działania programu
- `$ ./oitimetool ./nazwaprogramu < plik_testowy.in > wynik.out`  
aby uruchomić swój program w wirtualnym środowisku emulującym działanie procesora (w podobnym środowisku Jury będzie oceniał rozwiązania). Aby polecenie `./oitimetool` zadziałało, musisz być w folderze `/home/zawodnik/rozw`

## Testowanie

Aby przetestować swój program na testach przykładowych, możesz użyć skryptu `./ocen`. W tym celu:

- zapisz swoje rozwiązanie w folderze `/home/zawodnik/rozw` z nazwą odpowiadającą trzyliterowemu kodowi zadania i rozszerzeniem odpowiadającym językowi, w którym piszesz
- w terminalu wejdź do ww. folderu
- uruchom komendę

```
$ ./ocen ZADANIE
```

kompilującą i uruchamiającą rozwiązanie wybranego zadania na testach przykładowych.