

# 1 Wstęp

Podczas II etapu ocenie podlegają kody źródłowe programów, niniejszy dokument ma na celu usystematyzowanie zasad, którymi należy się kierować przy ich pisaniu.

## Można założyć, że:

- pliki wejściowe są poprawne,
- programy będą uruchamiane w systemie operacyjnym Linux, na komputerze z procesorem 486DX (lub lepszym),
- program będzie miał do dyspozycji 16MB pamięci, (jednak ta wartość dotyczy sumarycznego zapotrzebowania programu na pamięć, a więc zawiera rozmiaru kodu wykonywalnego, stosu, danych, sterty, itp.).

## Rozwiązania mogą:

- korzystać z biblioteki matematycznej (dla C/C++),
- korzystać z biblioteki STL (dla C++),
- otwierać do odczytu plik z danymi wejściowymi,
- otwierać do zapisu plik z danymi wyjściowymi.

## Rozwiązania powinny:

- kończyć się kodem wyjścia równym 0, każdy inny kod zakończenia programu jest równoznaczny błędowi wykonania,
- plik z rozwiązaniem powinien nosić nazwę `???.{c|cpp|pas}`, gdzie ??? jest trzyliterowym skrótem zadania, np. `abc.pas`, czy `abc.cpp`. Rozszerzenie pliku odpowiada wybranemu językowi programowania:

<code>.pas</code>	dla języka Pascal,
<code>.c</code>	C,
<code>.cpp</code>	C++,

- być umieszczone bezpośrednio w katalogu `/home/oi/rozw/` (w przypadku korzystania z systemu Linux) lub `C:\oi\rozw` (dla DOS), oznacza to, że rozwiązania zadania o kodzie `abc`, będzie poszukiwane pod nazwami:

- `/home/oi/rozw/abc.{c|cpp|pas}` oraz
- `C:\oi\rozw\abc.{c|cpp|pas}`

i jeśli nie zostanie znalezione pod żadną z tych nazw, uznawane jest to za brak rozwiązania. Na komputerze może być umieszczone tylko jedno rozwiązanie każdego zadania.

## Rozwiązanie zadań nie mogą:

- tworzyć nowych procesów, czy wątków,
- uruchamiać innych programów,
- korzystać z zewnętrznych bibliotek (oprócz biblioteki matematycznej dla C/C++), np. `crt`, `system` itp., dotyczy to również bibliotek dołączanych przez opcje kompilacji,
- używać funkcji sieciowych (np. `socket`, `gethostbyname`, itp),

- otwierać plików (oprócz wymienionych w poprzednim punkcie), w szczególności zabronione jest tworzenie plików tymczasowych,
- wykorzystywać większej ilości pamięci niż 16MB,
- kod źródłowy rozwiązania nie może przekraczać 100kb,
- naruszać bezpieczeństwa systemowego,
- oczekiwać na interakcję użytkownika.

Naruszenie powyższych zasad może mieć różne konsekwencje, od nieprzewidywalnego działania programu do dyskwalifikacji włącznie.

## 2 Kompilowanie rozwiązań

Rozwiązania napisane w języku Pascal będą kompilowane przy użyciu kompilatora *Free Pascal Compiler 1.0.4*, następującym poleceniem:

```
ppc386 rozwiązanie.pas
```

Rozwiązania napisane w języku C będą kompilowane przy użyciu kompilatora *GCC 2.95.x*, następującym poleceniem:

```
gcc -lm rozwiązanie.c
```

Rozwiązania napisane w języku C++ będą kompilowane przy użyciu kompilatora *GCC 2.95.x*, następującym poleceniem:

```
g++ -lm rozwiązanie.cpp
```

Użyty język programowania rozpoznawany jest na podstawie rozszerzenia pliku. Nie jest możliwy inny sposób kompilacji (np. przy pomocy plików *Makefile*).

## 3 Na co należy zwrócić uwagę?

- Ocenie podlegają kody źródłowe, które następnie będą kompilowane, stąd ważne jest by zawodnik zadbał o to, żeby kompilacja zakończyła się sukcesem, szczegółowe zapisy dotyczące kompilacji można znaleźć w rozdziale “Kompilowanie rozwiązań”.
- Programy powinny zwracać kod wyjścia równy 0, jest to ważne zwłaszcza w C/C++, gdzie konieczne jest dodanie `return 0` w funkcji `main`. Gdy program zwraca inny kod wyjścia, traktowane jest to jako błąd wykonania.
- Poprawne nazwy programów, plików z danymi wejściowymi, wyjściowymi — rozwiązania testowane są automatycznie, jeśli program/dane nie zostaną znalezione pod nazwami podanymi w specyfikacji treści zadania, to kończy się to komunikatem: brak rozwiązania, błąd wykonania czy brak odpowiedzi. **Uwaga! Ponieważ rozwiązania będą testowane w środowisku, w którym ważne są duże/małe litery w nazwach plików, stąd wszystkie nazwy plików powinny być pisane małymi literami, dokładnie tak, jak zostało podane w specyfikacji zadania.**
- Zgodność odpowiedzi generowanych przez rozwiązania ze składnią podaną w treści zadania, jedynymi dopuszczalnymi odstępstwami są:
  - białe znaki (spacja, znak tabulacji) na końcu wiersza, (np. `'0_1_'` zamiast `'0_1'`),
  - białe znaki (spacja, znak tabulacji, znak końca linii) na końcu pliku.
- Wypisywanie informacji na standardowe wyjście, jakkolwiek nie jest zabronione, może czasami znacząco wpływać na czas wykonania programu.