

Praktyka programowania 2012/2013

# Instrukcja laboratoryjna cz.0

Wprowadzenie

Prowadzący: Tomasz Goluch

Wersja: 2.1

## I. Warunki zaliczenia

*Cel: Zapoznanie studentów z warunkami zaliczenia części laboratoryjnej przedmiotu.*

Najważniejsze informacje dotyczące zaliczenia części laboratoryjnej przedmiotu:

- Niniejsza instrukcja jest dostępna pod adresem: [http://kaims.pl/~goluch/lab/InstrLabPPcz0\\_2012-13](http://kaims.pl/~goluch/lab/InstrLabPPcz0_2012-13) – proszę sprawdzić czy nie pojawiła się **zaktualizowana** wersja.
- Obecność na laboratoriach jest **obowiązkowa**.
- Pierwsze zajęcia mają charakter **wprowadzający**.
- Odbędzie się **osiem** laboratoriów (**jedno** wprowadzające i **siedem** punktowanych), każde po **dwie** godziny zajęciowe.
- Godzina zajęciowa trwa **45 min**.
- Na każde z zajęć składa się od **trzech** do **sześciu** programów do **indywidualnego** napisania.
- Wraz z rozpoczęciem nowych zajęć mija termin oddawania wcześniejszych, zaległych zadań laboratoryjnych.
- Na każdych zajęciach można otrzymać do **5 pkt**.
- Jeśli ktoś nie zdąży może oddać resztę zadań na kolejnych zajęciach ale za **połowę pkt**.
- W sumie można uzyskać **7 x 5 = 35 pkt**, co stanowi ok. **117%** części oceny laboratoryjnej.
- Próg zaliczenia laboratorium to **50 % (15 pkt.)**. Jednak do zaliczenia całego przedmiotu należy posiadać ponad **50%** sumy punktów z wszystkich trzech części (wykład, laboratorium, projekt).
- Wyniki laboratorium **podgrup: 04 A i B** dostępne są na stronie domowej: [www.kaims.pl/~goluch](http://www.kaims.pl/~goluch) w zakładce: Dla studentów → Semestr letni 2012/2013 → Praktyka programowania → laboratorium → Wyniki laboratorium grupy: [04 rok. 2012/13.](#)

## II. Kryteria oceny

*Cel: Zapoznanie studentów z najważniejszymi elementami mającymi wpływ na ocenę.*

Informacje dotyczące oceny z części laboratoryjnej przedmiotu:

- Laboratorium ma na celu zweryfikowanie wiedzy pozyskanej na wykładzie co oznacza, że jest to **praca indywidualna**, aczkolwiek w przypadku wątpliwości można zadawać pytania prowadzącemu.
- Podstawą zaliczenia programu jest **umieszczenie** kodu na platformie SPOX w godzinach **trwania** laboratorium.
- Programy mogą być zatwierdzone automatycznie przez platformę SPOX albo indywidualnie przez prowadzącego. Jednak w obydwu przypadkach kod **musi** zostać zamieszczony na platformie SPOX.
- Ocenie będzie podlegał tylko **ostatni zaakceptowany** kod, w przypadku oceny indywidualnej odpowiednio będzie to **ostatnio umieszczony** kod o dowolnym wyniku.
- Kod musi się **kompilować, wykonywać i działać** poprawnie.
- Na laboratorium dostępne są następujące środowiska programistyczne:

- Microsoft Visual Studio 2008 i 2010<sup>1</sup>
- Dev-C++ 4
- Code::Blocks
- Ideone – kompilator online i narzędzie do debugowania dostępne pod adresem: [www.ideone.com](http://www.ideone.com)
- Czytelność kodu (kod powinien być zrozumiały nie tylko dla autora ale i dla prowadzącego laboratorium).
- Brak powtórzeń kodu (stosowanie pętli oraz funkcji).
- Stosowanie przejrzystego stylu programowania (formatowanie kodu, odstępy, konwencja nazewnicza) – nie wymaga się konkretnego stylu jednak ważne aby był on jednolity i najlepiej zgodny z którymś ze standardów (przykładowo środowisko programistyczne Microsoft Visual Studio zapewnia automatyczne formatowanie kodu).
- Stosowanie komentarzy tylko w ostateczności, najlepiej gdyby kod ich nie wymagał (można to osiągnąć wykorzystując odpowiednie nazwy zmiennych i stałych, nazwy funkcji, typy numeryczne, dyrektywy preprocesora – `#define` oraz słowa kluczowe – `typedef`).

### III. Platforma internetowa SPOX

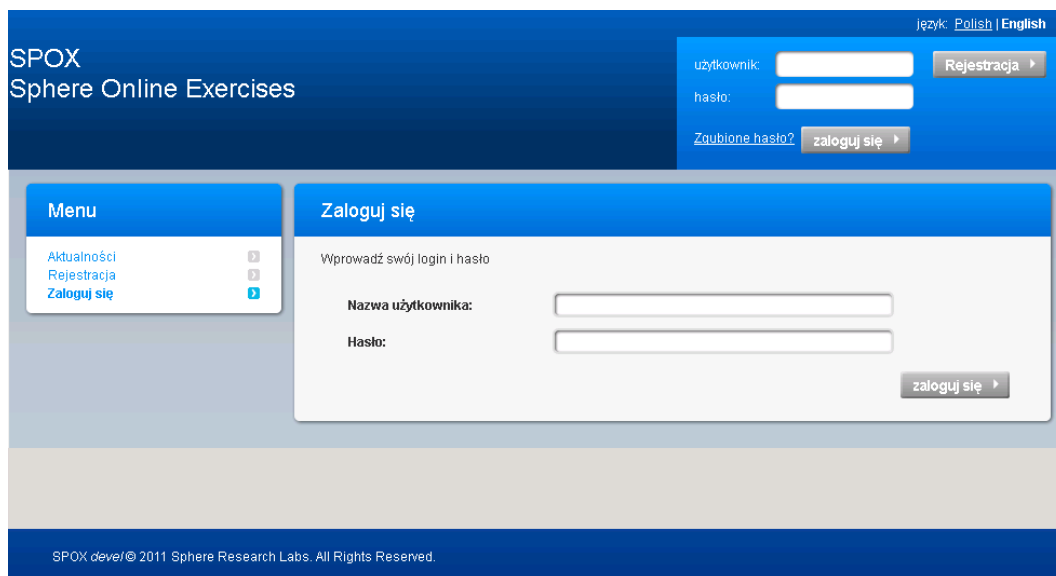
*Cel: Zapoznanie studentów z procedurą zakładania konta oraz obsługi platformy SPOX..*

Zasady zakładania konta i obsługi internetowej platformy SPOX:

- Odnośnik do kursu danej grupy dostępny na stronie domowej: [www.kaims.pl/~goluch](http://www.kaims.pl/~goluch) w zakładce: Dla studentów → Semestr letni 2012/2013 → Praktyka programowania →laboratorium →Strona laboratorium grupy: → **właściwa podgrupa**.
- Strona startowa została przedstawiona na rys.1, w pierwszym kroku należy wybrać opcję → Rejestracja.

---

<sup>1</sup> Każdy student może otrzymać środowisko programistyczne Microsoft Visual Studio za pomocą programu Microsoft DreamSpark® (dawniej MSDNAA). Więcej szczegółów dostępnych jest na stronie programu: <http://www.eti.pg.gda.pl/pracownicy/oprogramowanie/Aby.htm>



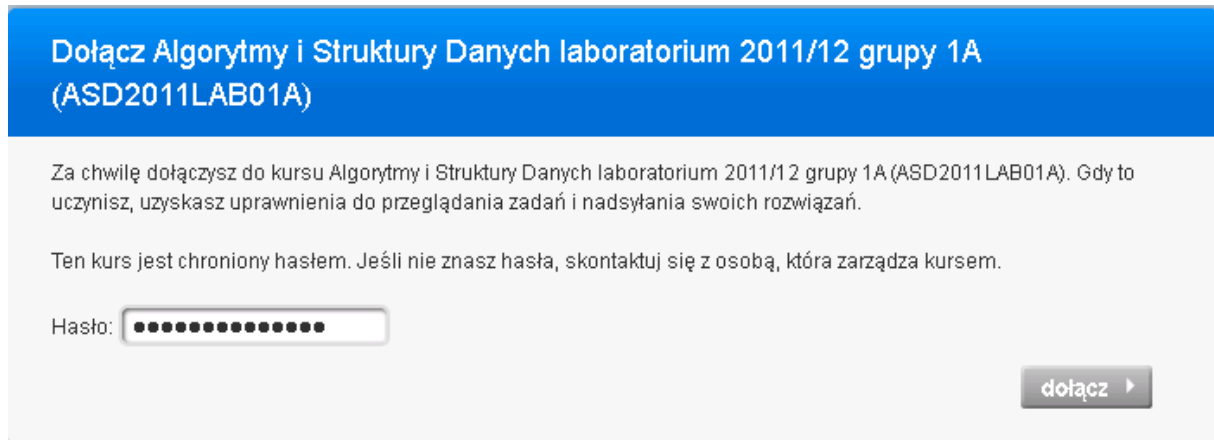
Rysunek 1 - Strona startowa platformy SPOX.

- Okno rejestracji nowego użytkownika przedstawiono na rys. 2

Rysunek 2 - Okno rejestracji nowego użytkownika.

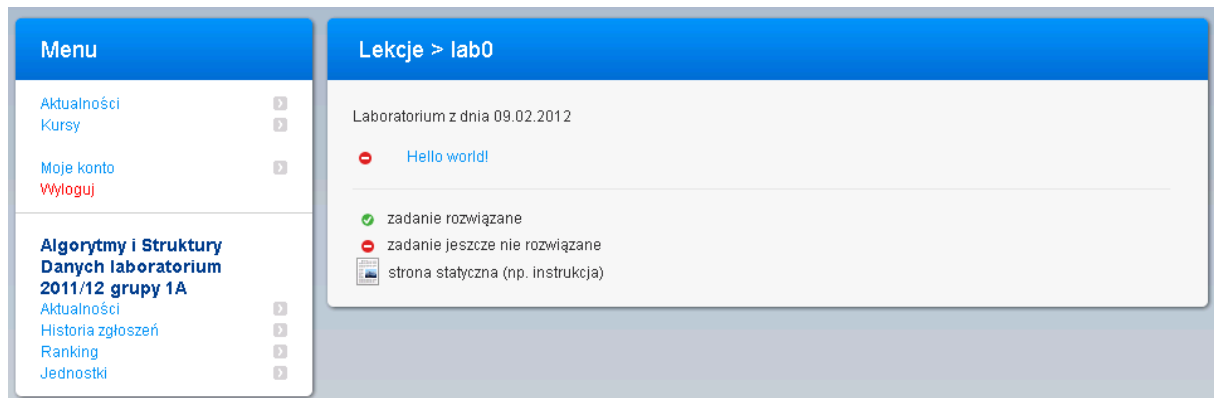
- **UWAGA !!!** – jako nawę użytkownika proszę wpisać **TYLKO NUMER INDEKSU**, inne nazwy będą ignorowane i usuwane!
- Adres email to dowolny adres jakim użytkownik dysponuje, jest wymagany w celu potwierdzenie rejestracji.
- W celu dołączenia do kursu należy wprowadzić hasło które otrzymają państwo od prowadzącego, rys 3.

- W przypadku problemów z odnalezieniem strony kursu proszę jeszcze raz skorzystać z odnośnika na stronie domowej: [www.kaims.pl/~goluch](http://www.kaims.pl/~goluch) w zakładce: Dla studentów → Semestr letni 2011/2012 → Algorytmy i Struktury Danych → Strona laboratorium grupy: → [właściwa podgrupa](#).



Rysunek 3 - Okno pozwalające na wprowadzenie hasła dostępu do kursu.

- Po zalogowaniu proszę wybrać zadanie testowe „**Hello world!**” z laboratorium wprowadzającego (**lab0**): Jednostki → Lekcje → lab0 → Hello world! – rys 4.



Rysunek 4 - Okno z zawartością laboratorium wprowadzającego.

- Następnie proszę wybrać dostępne zadanie (Hello world!).

# Hello world!

Nadane rozwiązania: 0

## Przykład

Napisz program, który wyświetli napis Hello world!

## Rozwiązanie

```
#include <iostream>

using namespace std;

int main()
{
    cout << "Hello world!";

    return 0;
}
```

- [Zobacz wynik działania programu.](#)

Rysunek 5 - Okno z treścią zadania.

- Proszę zapoznać się z treścią zadania rys. 5. Klikając na odnośniku „Zobacz wynik działania programu” można skompilować i sprawdzić poprawność programu.
- W celu automatycznej weryfikacji kodu proszę umieścić go w oknie „Prześlij rozwiązanie”, wybrać dostępny kompilator i wybrać opcję „wyślij” rys.6.

## Prześlij rozwiązanie

Wklej kod źródłowy:

```
1 #include <iostream>
2
3
4
5 using namespace std;
6
7
8
9 int main()
10
11 {
12
13     cout << "Hello world!";
14
15     return 0;
16
17 }
```

Język: C++ (g++ 4.0.0-8) ▾

Wybierz plik:

Przełóżaj...

wyślij ▶

Rysunek 6 - Okno pozwalające na przesłanie rozwiązania.

- W celu automatycznej weryfikacji kodu proszę umieścić go w oknie „Prześlij rozwiązanie”, wybrać dostępny kompilator i wybrać opcję „wyślij” rys.6.

| Historia zgłoszeń |                         | Ranking                   |        |       |                        |
|-------------------|-------------------------|---------------------------|--------|-------|------------------------|
| Id                | Data                    | Wynik                     | Punkty | Język |                        |
| 6834              | 12 lutego 2012 16:08:02 | zaakceptowano (10 Punkty) | 10     | C++   | <a href="#">edytuj</a> |

- W przypadku braku błędów w oknie Historii zgłoszeń powinien pojawić się wynik zadania w kolorze zielonym. W przeciwnym przypadku powinien pojawić się opis błędu.
- Proszę nie zwracać uwagi na liczbę punktów przyznawanych przez platformę. W szczególności nie mają one nic wspólnego z punktami z zaliczenia laboratorium które są indywidualnie przyznawane przez prowadzącego.
- Niektóre rodzaje zadań nie podlegają automatycznej weryfikacji i są ręcznie sprawdzane przez prowadzącego. Należy wtedy zaprezentować prowadzącemu działający skompilowany kod i dodatkowo umieścić go na platformie.
- Po zakończeniu zajęć proszę **WYŁĄCZYĆ KOMPUTER !!!**