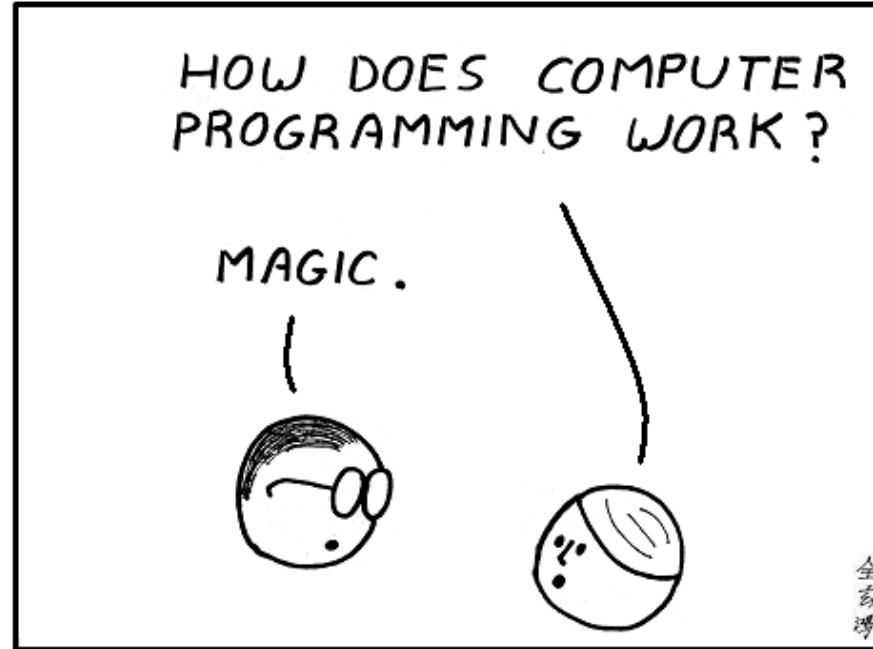


Języki Programowania: C++



http://www.saltbox.com/img/under_the_hood.png

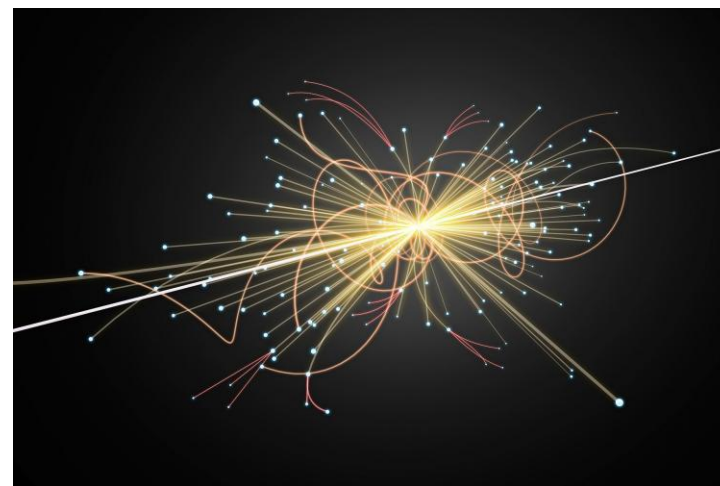
dr hab. inż. Małgorzata Janik, prof. PW

Any sufficiently advanced technology is indistinguishable from magic.

Arthur C. Clarke

O mnie...

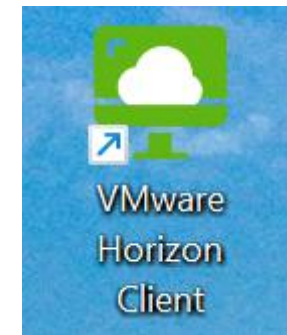
- **Profesor w Zakładzie Fizyki Jądrowej Wydziału Fizyki PW**
 - **dr hab. inż. Małgorzata Janik**
- **Kończyłam wydział Fizyki (fizyka komputerowa) i Wydział MiNI (computer science) PW**
- **Prowadzę badania z Fizyki Cząstek w eksperymencie ALICE na Wielkim Zderzaczu Hadronów w CERN**
 - **Materia w stanie “plazmy kwarkowo-gluonowej”**
 - **Korelacje między produkowanymi cząstkami**



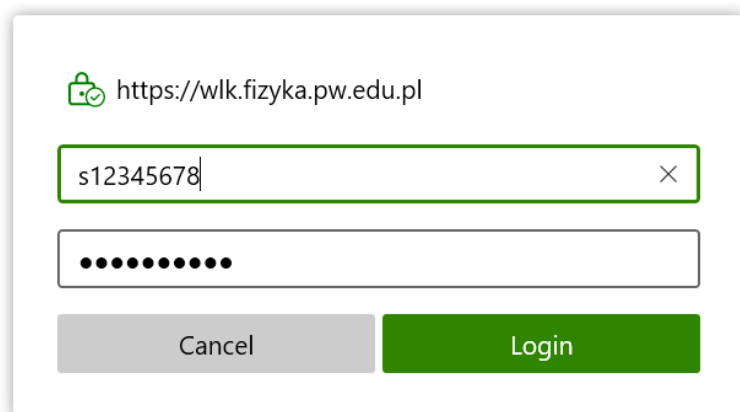
Na czym pracujemy? WLK

Jak się zalogować na komputer? → Windows →
Login: STUD\0USOS_id . Hasło do usosa.

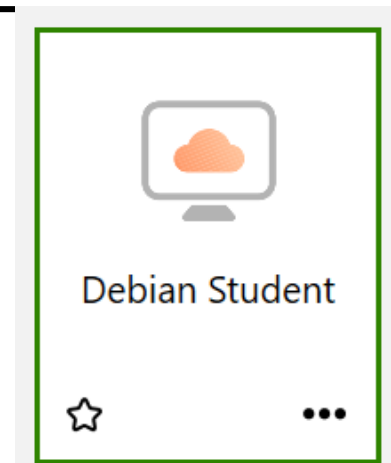
WLK: VMWare Horizon Client →



wlk.fizyka.pw.edu.pl → Login: s0USOS_id →

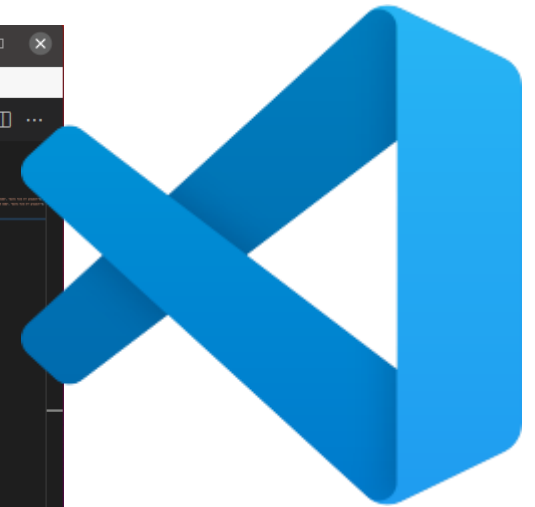
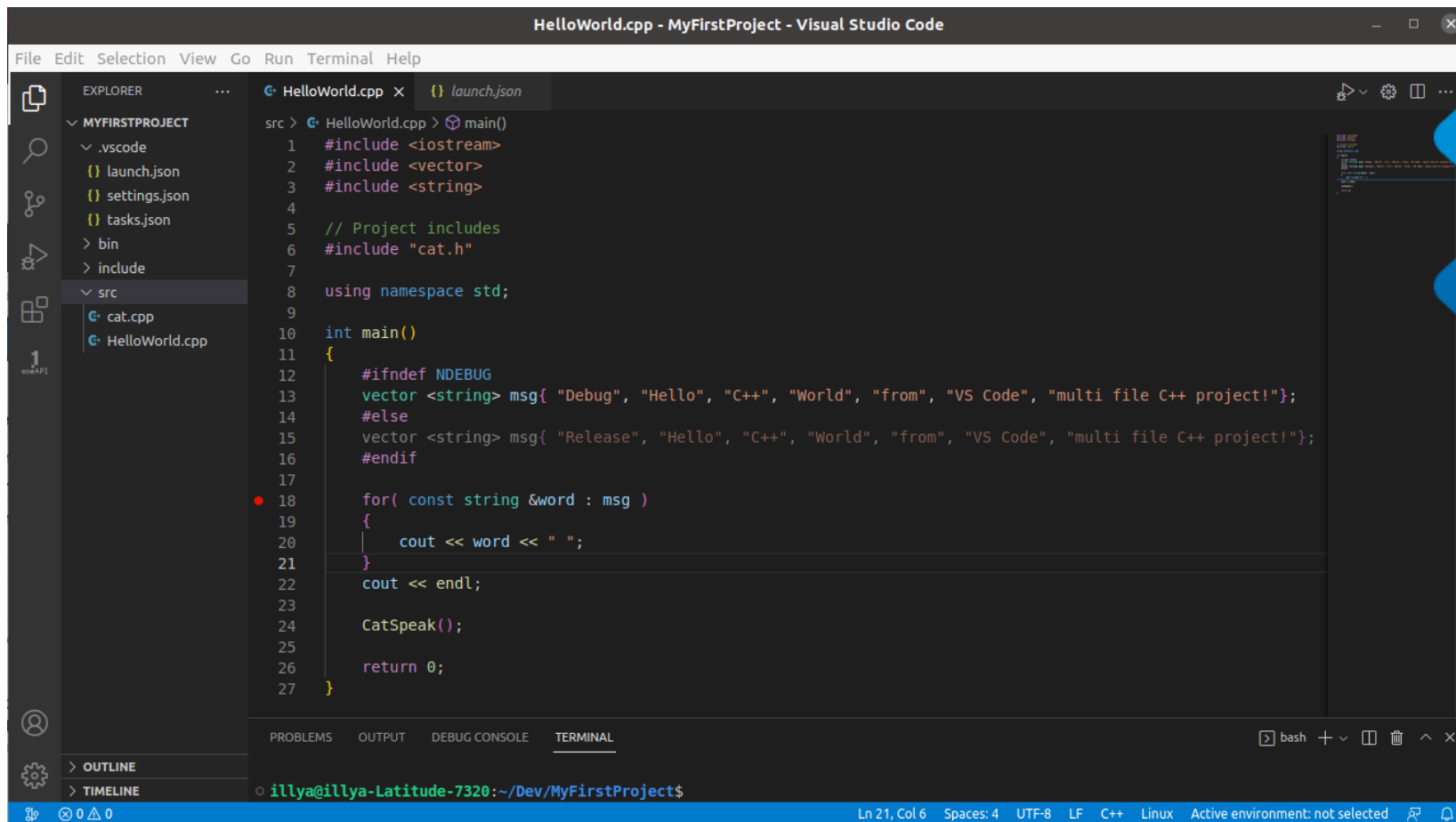
The image shows a web browser login form for the URL https://wlk.fizyka.pw.edu.pl. It features a text input field containing "s12345678", a password field with masked characters, and two buttons: "Cancel" and "Login".

Debian



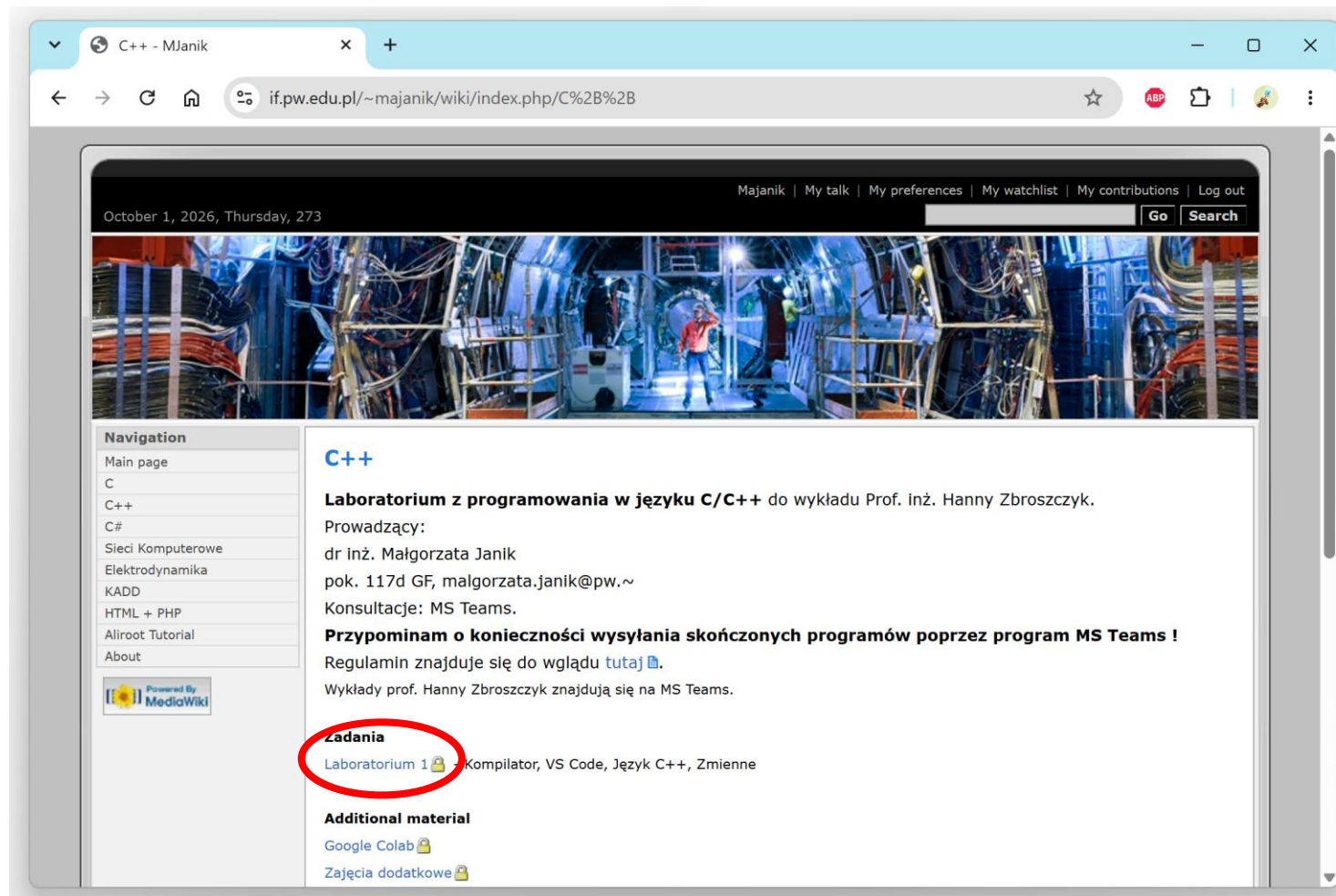
Na czym pracujemy? VS Code

Applications → Programming → Visual Studio Code



https://www.if.pw.edu.pl/~majanik/wiki

Applications → Internet → Firefox



Warunki zaliczenia

- **Wykład – 15 wykładów x 2h co tydzień**
Nieobowiązkowy. Dla osób uczęszczających na wykłady (możliwa 1 nieobecność), będzie możliwość uzyskanie dodatkowych 10 pkt.
- **Projekt: oddawany na ostatnich zajęciach laboratoryjnych**
50 pkt
- **Laboratoria: zajęcia punktowane**
13 x 5 pkt = 65 pkt
- **Warunek zaliczenia:**
 - **Zaliczony project (> 25 pkt)**
 - **Zaliczone zajęcia (> 35 pkt)**
- **Możliwe zaliczenie eksternistyczne.**

Ilość punktów	Ocena
≥ 60	3
≥ 72	3.5
≥ 84	4
≥ 96	4.5
≥ 108	5

Laboratoria

- przewidzianych jest 15 zajęć laboratoryjnych (w tym 13 punktowanych, 1 dzisiaj, 1 projektowe – oddanie projektów)
- obecność jest obowiązkowa (możliwe są maksymalnie 2 nieobecności);
- zajęcia trwają 90 minut, odbywają się bez przerwy;
- **Laboratoria: zajęcia punktowane (13 zajęć)**
 - za każde zadanie można otrzymać 0-5 pkt
 - w trakcie pisania programu wolno korzystać z napisanych przez siebie programów oraz zasobów Internetu; nie można korzystać z programów innych studentów ani komunikatorów
 - napisany w trakcie trwania laboratorium program należy oddać na tych samych zajęciach
 - za skończenie programu po zajęciach (w domu) i przedstawieniu go w kolejnym tygodniu można uzyskać dodatkowy punkt (ale maksymalnie 4 pkt za program)
 - przynajmniej jeden, a maksymalnie dwa programy będą pisane przez dwa, niekoniecznie następujące po sobie zajęcia;
 - w przypadku nieobecności studenci są zobowiązani do zrealizowania materiału we własnym zakresie i przedstawienia rozwiązania najdalej 2 tygodnie po nieobecności
 - AI: jeśli prowadzący nie zakomunikuje inaczej podczas danych konkretnych laboratoriów, użycie AI jest zabronione (dotyczy to również kończenia program w domu)

Projekty

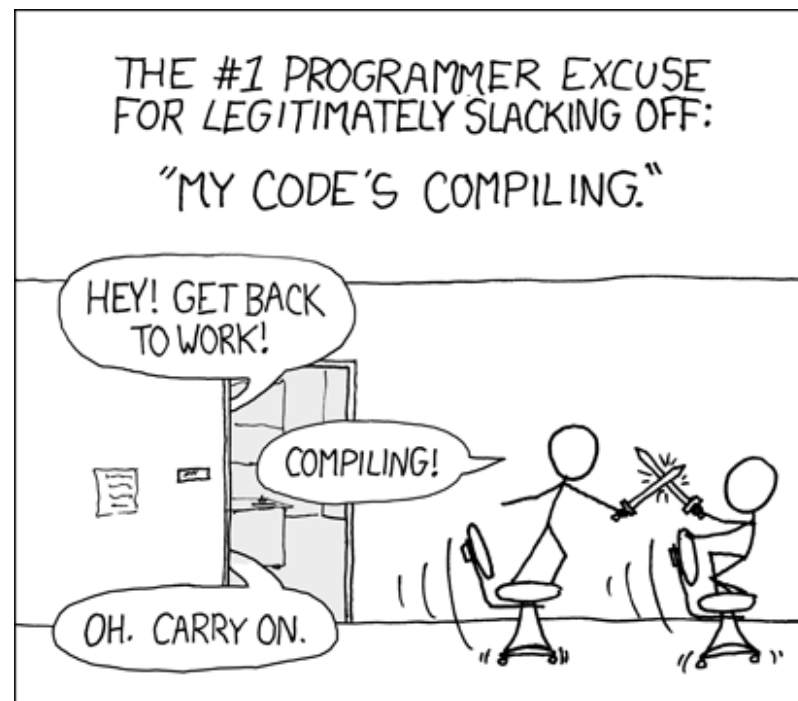
- Zajęcia kończą się oddaniem indywidualnego projektu, którego tematyka będzie indywidualnie ustalana z prowadzącym (zgłoszenie tematu **do 6 grudnia**).
- Zadanie projektowe musi zostać oddane przed zakończeniem semestru (maksymalnie w trakcie ostatnich zajęć w semestrze)
- próby niesamodzielnej pracy będą skutkowały niezaliczeniem projektu oraz brakiem możliwości jego poprawy - **niezaliczeniem przedmiotu!**;
- napisany projekt należy przesłać przed końcem semestru na adres e-mailowy prowadzącego / **GitHub**;
- użycie AI możliwe (chyba, że od początku student wyrazi chęć napisania samodzielnie prostrzego projektu)
 - nadal trzeba wyjaśnić, co dokładnie dzieje się w kodzie
- program będzie oceniany w skali 0-50 pkt

Komendy systemu linux

ls (list) - wyświetla zawartość bieżącego katalogu lub katalogu podanego jako parametr.
cd (change directory) - wchodzi do katalogu, np. **cd katalog1**
cd .. - wchodzi do katalogu wyżej
mkdir (make directory) - do tworzenia katalogów. Przykład: **mkdir nazwa_katalogu**
cp (copy) - do kopiowania plików i katalogów. Przykłady: **cp plik1 plik2**
cp -r - kopiuje katalog wraz z zawartością np. **cp -r katalog1 katalog2**
***** - gwiazdka zastępuje dowolny ciąg znaków np.:
cp * alfa/ - kopiuje wszystkie pliki z bieżącego katalogu do katalogu alfa
mv (move) - przenosi plik/pliki, służy też do zmiany nazwy pliku lub katalogu.
mv plik1 plik2 - zmienia nazwę plik1 na plik2
rm (remove) - usuwa pliki. Przykład:
rm plik1 - usuwa plik1
rm * - usuwa wszystkie pliki z bieżącego katalogu (należy używać bardzo ostrożnie - sprawdzić, czy rzeczywiście chcemy wszystko skasować).
rm -r - usuwa cały katalog razem z zawartością
more - pozwala na przeglądanie danych (plików, komunikatów poleceń) ekran po ekranie.
kate plik.txt - uruchamia edytor kate tworząc plik plik.txt
cat - podobnie do polecenia 'more' pokazuje zawartość pliku ale nie zatrzymuje się ekran po ekranie tylko wyświetla od razu całość.

Zadanie właściwe

https://www.if.pw.edu.pl/~majanik/data/JP/2026/Zajecia_1_CPP.html



<http://imgs.xkcd.com/comics/compiling.png>