

W wyniku wykonania poniższych zadań powinien powstać **program**, którego kody źródłowe powinny zostać przesłane na adres antyplagiatu (informacja w ramce na końcu zadania).

1. Wykorzystując kod z poprzednich zajęć napisz program, który będzie wyświetlał aktualnie zalogowanych w systemie użytkowników. Program musi spełniać następujące warunki:
 - a. wywołany bez przełączników wyświetli w kolejnych liniach loginy użytkowników, np.:

```
jkwalski
```

- b. wywołany z przełącznikiem **-h** (wykorzystaj **getopt**) wyświetli w kolejnych liniach loginy użytkowników i hosty, z których się zalogowali, np.:

```
jkwalski (89.64.110.125)
```

- c. wywołany z przełącznikiem **-g** wyświetli w kolejnych liniach loginy oraz nazwy wszystkich grup, do których dani użytkownicy należą (nie ograniczamy się do grupy głównej, używamy funkcji **getgrouplist**, kontrolujemy przydzielanie i zwalnianie pamięci), np.:

```
jkwalski [users, virtualserver]
```

- d. wywołany z obydwoma przełącznikami wyświetli w kolejnych liniach loginy, hosty i nazwy wszystkich grup, do których dani użytkownicy należą,

```
jkwalski (89.64.110.125) [users, virtualserver]
```

- e. pobieranie listy grup, do której należy użytkownik o podanym **UID**, powinno być zaimplementowane jako niezależna funkcja, która będzie umieszczona osobnym pliku.

Z pliku zawierającego naszą funkcję zwracającą grupy użytkownika należy stworzyć bibliotekę statyczną. Na koniec napisz prosty **Makefile**, który będzie automatyzował kompilację i konsolidację programu i tworzenie biblioteki.

Przed wysłaniem plików źródłowych na antyplagiat ich nazwy zmieniamy na:

- **numer_indeksu.ps.lab03.static.main.c** (czyli np. 66666.ps.lab03.static.main.c),
- **numer_indeksu.ps.lab03.static.lib.c** (czyli np. 66666.ps.lab03.static.lib.c),
- **numer_indeksu.ps.lab03.static.Makefile** (czyli np. 66666.ps.lab03.static.Makefile).

2. Bazując na wcześniejszym programie napisać jego nową wersję, w której część kodu zostanie umieszczona w bibliotece współdzielonej. Zamiast bibliotek statycznej musimy stworzyć bibliotekę współdzieloną. Działanie programu powinno być identyczne jak wersji wcześniejszej. Program próbuje załadować dynamicznie (za pomocą funkcji biblioteki **dlfcn**) przygotowaną wcześniej bibliotekę współdzieloną i zaimportować udostępnianą przez nią funkcję. Jeżeli zostanie uruchomiony z odpowiednimi przełącznikami powinien wyświetlić rozszerzoną informację o zalogowanych użytkownikach (używamy w tym celu m.in. funkcję z załadowanej biblioteki). W przypadku braku wymaganej funkcji (brak biblioteki lub brak funkcji w bibliotece) program powinien wyświetlić informację o tym braku i kontynuować działanie tak, jakby został uruchomiony bez przełączników. Napisz prosty **Makefile**, który będzie automatyzował kompilację i konsolidację programu i biblioteki.

Przed wysłaniem plików źródłowych na antyplagiat ich nazwy zmieniamy na:

- **numer_indeksu.ps.lab03.shared.main.c** (czyli np. 66666.ps.lab03.shared.main.c),
- **numer_indeksu.ps.lab03.shared.lib.c** (czyli np. 66666.ps.lab03.shared.lib.c),
- **numer_indeksu.ps.lab03.shared.Makefile** (czyli np. 66666.ps.lab03.shared.Makefile).

Kody źródłowe (6 plików) po oddaniu prowadzącemu zajęcia laboratoryjne muszą zostać jako załączniki przesłane na adres pss1@zut.edu.pl (wysyłamy jeden mail z czterema załącznikami):

- pliki z kodami źródłowymi muszą mieć nazwę zgodne ze wzorcem podanym w treści zadania,
- mail musi zostać wysłany z poczty uczelnianej (domena **zut.edu.pl**),
- temat maila musi mieć postać: **PS IS1 999X LAB03**, gdzie 999X to numer grupy laboratoryjnej (np. PS IS1 321 LAB03),
- w pierwszych trzech liniach kodu źródłowego w komentarzach (każda linia komentowana osobno) musi znaleźć się:
 - informacja identyczna z zamieszczoną w temacie maila,
 - imię i nazwisko osoby wysyłającej maila,
 - adres e-mail, z którego wysłano wiadomość,np.:

```
// PS IS1 321 LAB03
// Jan Nowak
// nj66666@zut.edu.pl
```

- e-mail nie może zawierać żadnej treści (tylko załączniki).

Dostarczone kody programów będą analizowane pod kątem wykrywania plagiatów. Niewysłanie wiadomości, wysłanie jej w formie niezgodnej z powyższymi wymaganiami lub wysłanie pliku, który nie będzie się kompilował i uruchamiał, będzie traktowane jako brak programu i skutkowało otrzymaniem za niego oceny niedostatecznej.