

## Przetwarzanie współbieżne. Programowanie równoległe i rozproszone

### Laboratorium 2

#### Cel:

- przeprowadzenie pomiaru czasu CPU i zegarowego tworzenia procesów i wątków systemowych Linuxa
- nabycie umiejętności posługiwania się programami wykorzystującymi tworzenie wątków i procesów

#### Zajęcia:

1. Utworzenie podkatalogu roboczego w katalogu PR\_lab (np. lab\_2).
2. Skopiowanie ze strony WWW do katalogu roboczego pliku „fork\_clone.tgz”
3. Rozpakowanie plików, uruchomienie programów (dla osób, które nie zrealizowały części dodatkowej z lab\_1, plik libpomiar\_czasu.tgz zawiera zawartość katalogu *pomiar\_czasu* (na tym samym poziomie jak lab\_1 i lab\_2, jako podkatalogi PR\_lab) – utworzenie biblioteki pomiaru czasu polega tylko na wykonaniu *make* w katalogu *pomiar\_czasu*)
4. Uzupełnienie plików źródłowych o procedury pomiaru czasu i przeprowadzenie eksperymentu mierzącego czas tworzenia 1000 procesów i 1000 wątków (uwaga na optymalizacje kompilatora, który może usunąć operacje nie wywołujące żadnych efektów – jednym z możliwych testów jest sprawdzenie czasów realizacji dla wersji do debugowania i zoptymalizowanej i analiza różnic). Na jego podstawie obliczenie narzutu systemowego na tworzenie pojedynczego procesu i pojedynczego wątku (można także zaprojektować i przeprowadzić odrębny eksperyment mierzący czas realizacji funkcji *exec*)
5. Modyfikacja kodu programów, tak aby procedura wątku i program uruchamiany (za pomocą funkcji *exec*) przez nowo utworzony proces wypisywały na ekranie indywidualnie zaprojektowane dane (np. imię i nazwisko, numer procesu lub tp.)
6. Modyfikacja programu, tak aby bezpośrednio po sobie tworzyć dwa wątki do działania równoległego, które będą zwiększać w odpowiednio długiej pętli wartości dwóch zmiennych: jednej globalnej (lub statycznej), drugiej lokalnej (na stosie – przekazanej jako argument). Obserwacja zachowania zmiennych poprzez wypisanie ich wartości: w funkcjach wątków po zakończeniu pętli, w funkcji *main* (oczywiście po zakończeniu pracy obu wątków!)

#### Dalsze kroki dla podniesienia oceny:

1. Analiza wywołania funkcji *clone()* (m.in. przeglądanie strony podręcznika „*man clone*”) i taka jej modyfikacja, aby jej działanie pokrywało się z działaniem funkcji *fork()*.
2. Analiza znaczenia parametrów funkcji *clone()* - wskaźnik stosu, opcje współdzielenia zasobów – testowanie wariantów opcji, np.:
  1. statyczna alokacja dużej tablicy w procedurze wątku i sprawdzenie jaki trzeba dobrać stos, aby procedura mogła zostać zrealizowana
  2. próba otwarcia pliku w przypadku braku współdzielenia systemu plików
  3. itd., itp.

#### Warunki zaliczenia:

1. Obecność na zajęciach i wykonanie co najmniej kroków 1-5
2. Oddanie napisanego odręcznie sprawozdania z krótkim opisem zadania (cel, zrealizowane kroki, wnioski) oraz kodem źródłowym utworzonym przez siebie umieszczonym przy opisie odpowiadających kroków. Sprawozdanie powinno zawierać konkretne wyniki pomiarów czasu wykonania konkretnych operacji, a wnioski **porównanie wyników między sobą oraz z wynikami z poprzedniego laboratorium** – czasu wykonania pojedynczej operacji arytmetycznej i pojedynczej operacji wejścia/wyjścia, a także wnioski dotyczące zachowania zmiennych.