|  |  |  |
| --- | --- | --- |
| 9.10.2023 | Łukasz Oprych  Gr.5 ITE | Temat:  Procesy, Wątki, Fork/Clone |

Cel:

Zapoznanie się z procesami i wątkami w systemie operacyjnym Linux z wykorzystaniem funkcji fork() i clone().

Wykonanie ćwiczenia:

1. Utworzenie katalogu lab\_2 oraz skopiowanie do niego pliku fork\_clone.tgz z kodem źródłowym, które rozpakowano w katalogu oraz przygotowanie katalogu pomiar\_czasu z kodem z zajęć poprzednich, który wykorzystano do pomiarów czasu.

Struktura katalogowa:

Obraz zawierający zrzut ekranu, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

F

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

1. Uzupełnienie plików źródłowych o procedury pomiaru czasu

Wywołanie w clone.c oraz fork.c funkcji inicjuj\_czas() przed pętla, oraz drukuj\_czas() po wykonaniu pętli.

Program fork.c:

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

1. Następnie skompilowano oba programy za pomocą wywołania komendy make fork i make clone. Następnie wykonano 3 pomiary czasowe:

Dla wersji niezoptymalizowanej:

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

Oraz wersji zoptymalizowanej:

Obraz zawierający tekst, zrzut ekranu, Czcionka, numer

Opis wygenerowany automatycznie

1. Utworzono nowy plik program.c, gdzie zainicjowano dwa wątki, które inkrementowały 100000 razy dwie zmienne, globalną oraz lokalną i czekały na zakończenie swojego działania. Wartości zmiennych były przesyłane za pomocą wskaźnika do wątku. Na koniec mogliśmy również sprawdzić jakie finalnie były wartości zmiennej lokalnej i globalnej.

Wykonywana funkcja wątku:

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

Zainicjowanie stosów, wątków, funkcja oczekiwania wątków:

Obraz zawierający tekst, zrzut ekranu, oprogramowanie, Oprogramowanie multimedialne

Opis wygenerowany automatycznie

Wynik wykonania program.c

Wersja niezoptymalizowana:

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

Wersja zoptymalizowana:

Obraz zawierający tekst, zrzut ekranu, Czcionka

Opis wygenerowany automatycznie

Wnioski:

* Aby uzyskać równolegle działający program, możemy stworzyć procesy za pomocą funkcji fork(), oraz dodatkowe wątki za pomocą funkcji clone(). Tworzenie wątku zajmuje mniej czasu, niż tworzenie osobnego procesu zgodnie z uzyskanymi wynikami.
* Z wyników prowadzonych pomiarów, można zauważyć, że tworzenie procesów oraz wątków w programie w wersji zoptymalizowanej ma dłuższe czasy CPU niż w wersji niezoptymalizowanej, w teorii optymalizacja powinna skrócić czas wykonywania operacji. Nieoczekiwane wyniki mogły być z winy sprzętu bądź oprogramowania.
* W sporządzonych wynikach tworzenia dwóch wątków działających równolegle w program.c zmienna lokalna na koniec ma wartość, która jest oczekiwana po wykonaniu 100000 iteracji. Zmienna globalna za to w przypadku kodu niezoptymalizowanego ma wynik różny od oczekiwanego ze względu na współdzielenie pamięci przez oba wątki, powoduje to, że gdy pierwszy wątek pobiera wartość zmiennej gdy została zmieniona przez drugi wątek. W przypadku kompilowania programu używając optymalizacji zmienna globalna, osiąga już oczekiwane wartości.