

Temat zajęć	Procesy w systemie Windows
Zakres materiału	Tworzenie i wykonywanie procesów w systemie Windows

Materiał teoretyczny

- funkcje `CreateProcessA()`, `WaitForMultipleObjects()`, `GetExitCodeProcess()`, argumenty linii poleceń i kod zakończenia procesu

Treść zadania

Napisz w języku C program spełniający poniższe wymagania:

- program akceptuje dokładnie jeden argument wywołania – uruchomienie go z inną liczbą argumentów albo bez argumentów ma skutkować wyprowadzeniem na `stderr` informacji o błędzie i przerwaniem pracy z kodem zakończenia równym 11;
- otrzymany argument ma być ciągiem cyfr dziesiętnych – jeśli tak nie jest, program wyprowadza na `stderr` informację o błędzie i kończy pracę z kodem zakończenia równym 12;
- długość otrzymanego argumentu musi być mniejsza równa 40 – jeśli tak nie jest, program wyprowadza na `stderr` informację o błędzie i kończy pracę z kodem zakończenia równym 13;
- jeśli długość otrzymanego argumentu jest równa 1, program od razu kończy pracę z kodem zakończenia równym wartości otrzymanej cyfry, np. uruchomiony z argumentem „2” zwraca kod powrotu 2;
- jeśli długość otrzymanego argumentu jest równa 2, program od razu kończy pracę z kodem zakończenia równym wartości większej z otrzymanych cyfr, np. uruchomiony z argumentem „12” zwraca kod powrotu 2;
- program dzieli otrzymany argument na dwie równe części, a jeśli argument ma nieparzystą długość, na dwie części, z których druga jest o jeden znak dłuższa od pierwszej;
- program uruchamia dwa procesy potomne (funkcja `CreateProcessA()`), uruchamiając w nich ten sam program, który jest wykonywany przez rodzica i przekazuje im jako argument odpowiednio: do pierwszego dziecka – lewą część otrzymanego argumentu, do drugiego dziecka – prawą część otrzymanego argumentu;
- rodzic czeka na zakończenie obu procesów potomnych (funkcja `WaitForMultipleObjects()`), a następnie wypisuje na `stdout` trzy linie tekstu:
 1. swój PID, PID pierwszego zakończzonego dziecka, argument, z którym pierwsze dziecko zostało uruchomione i kod powrotu pierwszego dziecka;
 2. swój PID, PID drugiego zakończzonego dziecka, argument, z którym drugie dziecko zostało uruchomione i kod powrotu drugiego dziecka;

3. swój PID i większy z kodów powrotów obu dzieci (wartość tę należy wyprowadzić w tej samej kolumnie, co kody powrotu potomków);
- rodzic kończy pracę, zwracając kod powrotu równy większemu z kodów powrotu obu dzieci;
 - w efekcie w ostatniej linii wyjścia powinna znaleźć się wartość największej cyfry z łańcucha przekazanego do programu jako argument.

Uwaga: poprawna sekwencja operacji to *CreateProcessA()*, *CreateProcessA*, *WaitForMultipleObjects()*. Inne rozwiązanie dyskwalifikuje rozwiązanie.

Przykładowe wyjście z programu:

```
> zad6.exe 1234567
8080 5580          1 1
8080 5644         23 3
8080                3

6160 5792         45 5
6160 6028         67 7
6160                7

4968 8080         123 3
4968 6160        4567 7
4968                7
```

Uwaga! Kod źródłowy programu (1 plik) po zaprezentowaniu prowadzącemu zajęcia laboratoryjne musi zostać jako **załącznik** przesłany na adres son1@wi.zut.edu.pl:

- plik z kodem źródłowym musi mieć nazwę: numer_indeksu.so.lab05.c (np. 66666.so.lab06.c),
- plik musi zostać wysłany z poczty uczelnianej (domena [zut.edu.pl](mailto:son1@wi.zut.edu.pl)),
- temat maila musi mieć postać: SO IN1 99X LAB06
gdzie 99X to numer grupy laboratoryjnej (np. SO IN1 20B LAB06),
- 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 wiadomośćnp.:

```
// SO IN1 20B LAB06
// Jan Nowak
// nj66666@zut.edu.pl
```

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

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 zadania i skutkowało otrzymaniem oceny niedostatecznej.