NAZWA

sProgramForeground – rozpocznij program pierwszoplanowo

STRESZCZENIE

```
#include <fork.h>

 int sProgramForeground(char *progName, char *const args[],
P_S *p, int seq);
```

OPIS

sProgramForeground() na podstawie podanej nazwy polecenia/programu *progName*, sformatowanej na potrzeby funkcji **execvp**() listy argumentów *args* argumentów, struktury *p* zawierającej deskryptory plików oraz liczby całkowitej *seq* wskazującej kolejność łączenia potoków z procesami uruchamia zadane przez użytkownika polecenie bądź polecenia.

Funkcja po wywołaniu rozpoczyna działanie od stworzenia procesu dziecka przy pomocy **fork**() i przypisaniu jego PID do zmiennej. W przypadku, gdy jest to proces potomny (PID == 0), sprawdzana jest zmienna wskaźnikowa ***p**. Jeżeli zmienna nie jest NULLem, to funkcją **dup2**() łączone są standardowe wejścia i wyjścia procesów zgodnie z kolejnością opisywaną przez zmienną *seq*. Po zakończeniu łączenia zamykane są deskryptory. Następnie niezależnie od zawartości ***p** uruchamiane jest polecenie za pomocą **execvp**(*progName*, *args*).

W przypadku procesu rodzica funkcja sprawdza wpierw, czy wykorzystane zostały potoki oraz czy jest to ostatnie wywołanie pętli z nimi związanej (*p* != NULL && *seq* == *p* -> *size* - 2). Jeśli tak, to zamykane są kopie potoków. Następnie program odczekuje na zakończenie wszystkich procesów potomnych. W przypadku, gdy nie jest to ostatnie wywołanie pętli to program zwraca 0 czekając na kolejnę wywołanie. Natomiast jeśli potoki nie zostały wykorzystane to program po prostu czeka na zakończenie procesu potomnego.

ZWRACANA WARTOŚĆ

W razie powodzenia bądź odczekiwania na zakończenie łączenia potoków zwracane jest 0. W przypadku błędów, ze względu na ich kluczowość w działaniu “Skorupy”, cały program zakańcza pracę i wypisuje błąd.