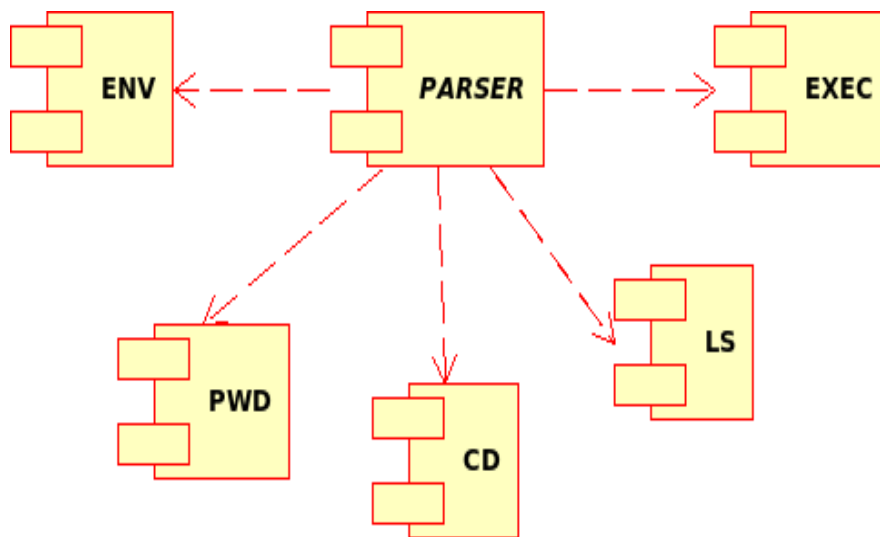


# Projekt Programistyczny – materiały dodatkowe

## Wymagania

1. Wbudowaną obsługę zmiennych środowiskowych, tzn. możliwość ustawiania i wyświetlania dowolnej zmiennej środowiskowej.
2. Wbudowane polecenie pwd wyświetlające aktualny katalog roboczy
3. Wbudowane polecenie cd pozwalające na zmianę katalogu roboczego. Polecenie powinno akceptować nazwy ścieżkowe. Wykonane bez argumentu powinno zmieniać katalog roboczy na katalog domowy użytkownika.
4. Wbudowane polecenie ls wyświetlające zawartość katalogu roboczego (jeżeli podane bez argumentów) lub dowolnego podanego jako argument.
5. Wbudowane polecenie ll odpowiadające poleceniu ls -l. Funkcjonalność jak w pkt 4.
6. Możliwość uruchomienia dowolnego programu poprzez podanie jego nazwy (ew. nazwy ścieżkowej).

## Proponowana struktura projektu - komponenty



## Parser

Analiza polecenia. Wywoływanie funkcji obsługujących poszczególne polecenia. Np.

- cmd\_env(char\* cmdline)
- cmd\_pwd(char\* cmdline)
- cmd\_cd(char\* cmdline)
- cmd\_ls(char\* cmdline)
- cmd\_exec(char\* cmdline)

Każda funkcja obsługuję jedną z funkcji (env – obsługa zmiennych środowiskowych, pwd – obsługa polecenia pwd, cd – obsługa polecenia cd itd.).

Proponowana lista poleceń rozpoznawanych przez parser:

- set zmienna=wartość
- unset zmienna
- echo zmienna
- pwd
- cd ścieżka
- cd
- ls
- ls ścieżka
- ll
- ll ścieżka
- exec ścieżka
- quit/exit

Algorytm działania parsera:

1. Wyświetl znak zachęty
2. Wczytaj linię
3. Rozpoznaj polecenie
4. Przekaż sterowanie do wybranej funkcji lub zakończ program (quit/exit)
5. Wróć do punktu 1

Standardowe funkcje przydatne przy pisaniu parsera: printf, scanf, fgets, strepy, strncpy, strcmp, strncmp, strok.

### ***Moduł env – manipulowanie zmiennymi środowiskowymi***

Działanie:

1. Rozpoznanie polecenia
2. Wydobywanie argumentów (nazwa, ew. wartość zmiennej)
3. Wywołanie właściwej funkcji bibliotecznej.

Przydatne funkcje standardowe: getenv, putenv, setenv, unsetenv, sscanf, printf, strok.

### ***Moduł pwd – wyświetlenie nazwy katalogu roboczego***

Działanie:

1. Wywołanie odpowiedniej funkcji bibliotecznej

Przydatne funkcje standardowe: getcwd, printf.

### ***Moduł cd – zmiana katalogu roboczego***

Działanie:

1. Sprawdzenie czy podany został argument.

2. Jeżeli argument jest to wywołanie funkcji bibliotecznej
3. Jeżeli nie ma argumentu to pobranie zawartości zmiennej środowiskowej HOME i wywołanie funkcji bibliotecznej.

Przydatne funkcje standardowe: chdir, sscanf, strok

Można wykorzystać funkcjonalność modułu env.

### ***Moduł ls – wyświetlenie zawartości katalogu***

Działanie:

1. Określenie typu polecenia (krótki lub długi format)
2. Sprawdzenie czy podany został argument.
3. Jeżeli brak argumentu to pobranie wartości zmiennej środowiskowej PWD
4. Otwarcie katalogu i kolejne odczytywanie pozycji w katalogu.
5. Jeżeli długi format to odczytanie atrybutów każdej pozycji.
6. Wyświetlenie informacji zgodnie z założonym formatem
7. Zamknięcie katalogu.

Przydatne funkcje standardowe: opendir, readdir, closedir, stat, sscanf, strcmp, strcat, strok, printf.

### ***Moduł exec – uruchomienie programu***

Działanie:

1. Wyodrębnienie nazwy uruchamianego programu z polecenia
2. Rozwidlenie procesu (fork)
3. Proces macierzysty czeka na zakończenie potomka, a później kontynuuje działanie.
4. Proces potomny uruchamia podany program (exec).

Przydatne funkcje standardowe: sscanf, strok, fork, wait, waitpid, jedna z funkcji z rodziny exec.