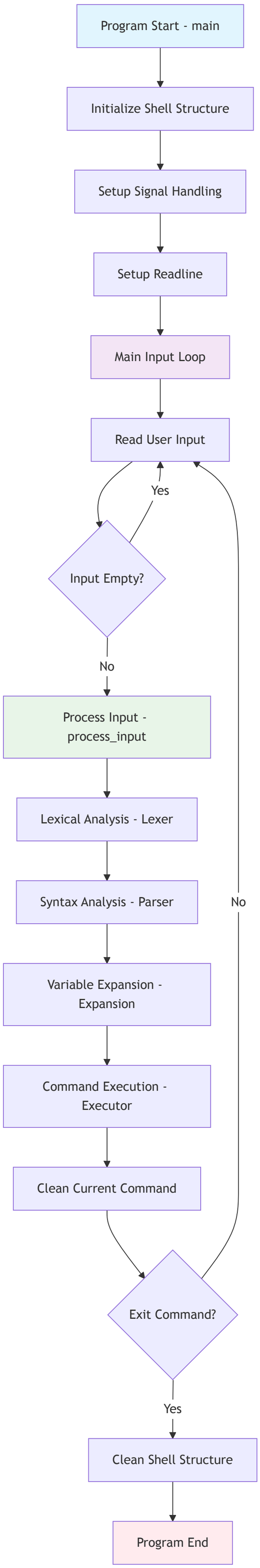
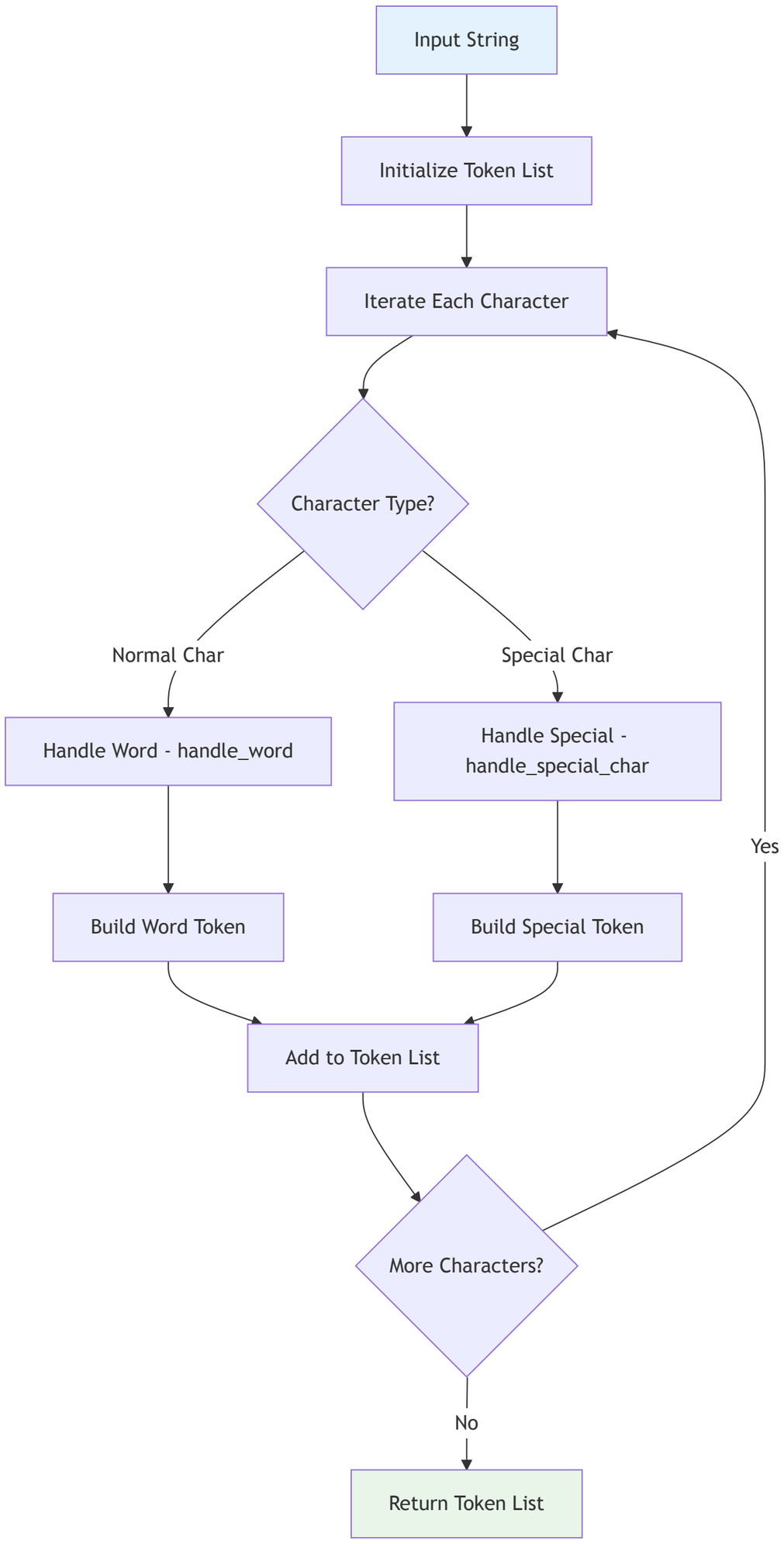
Minishell Project Flowchart

Overall Architecture Flowchart

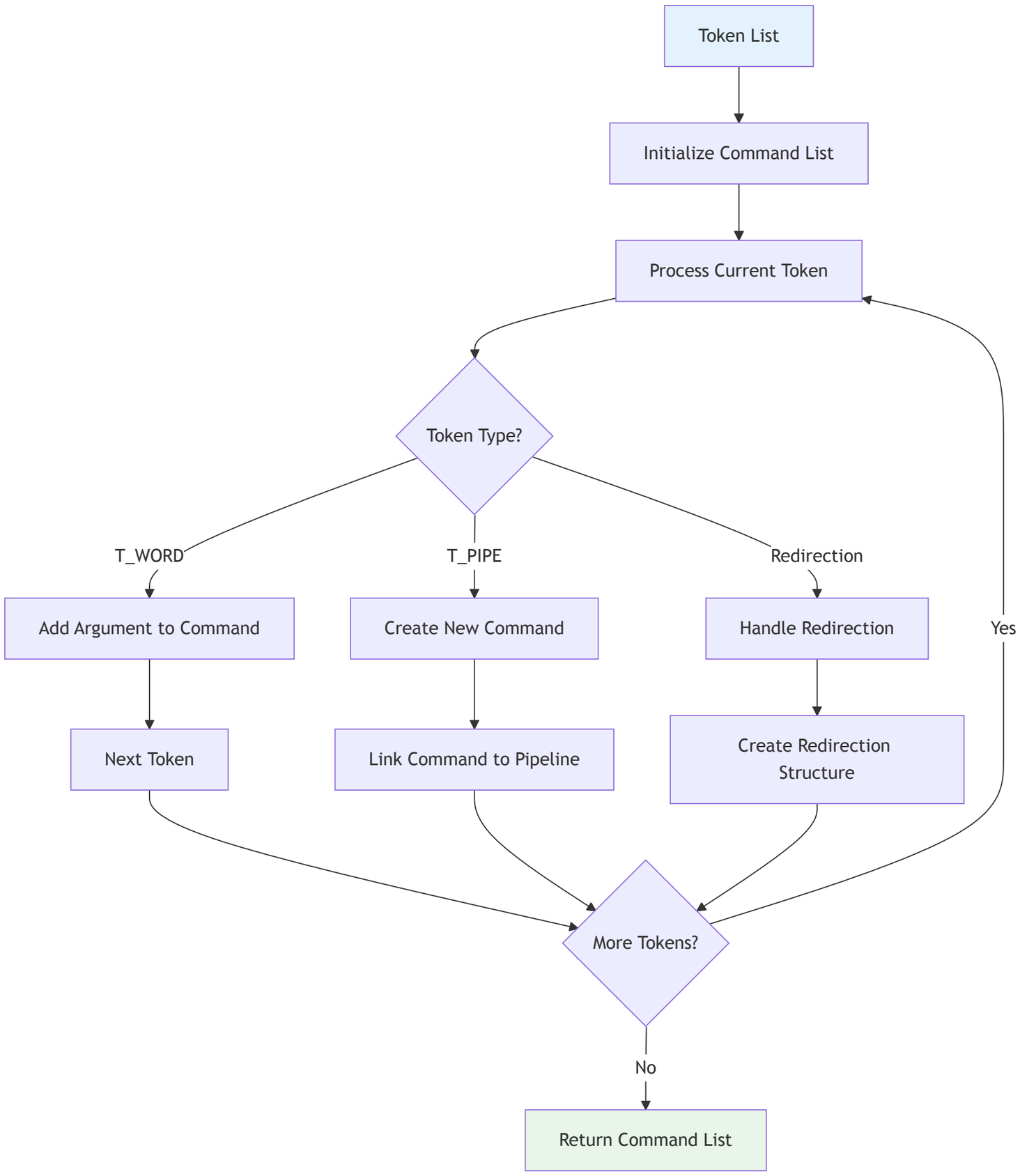


Detailed Module Flowcharts

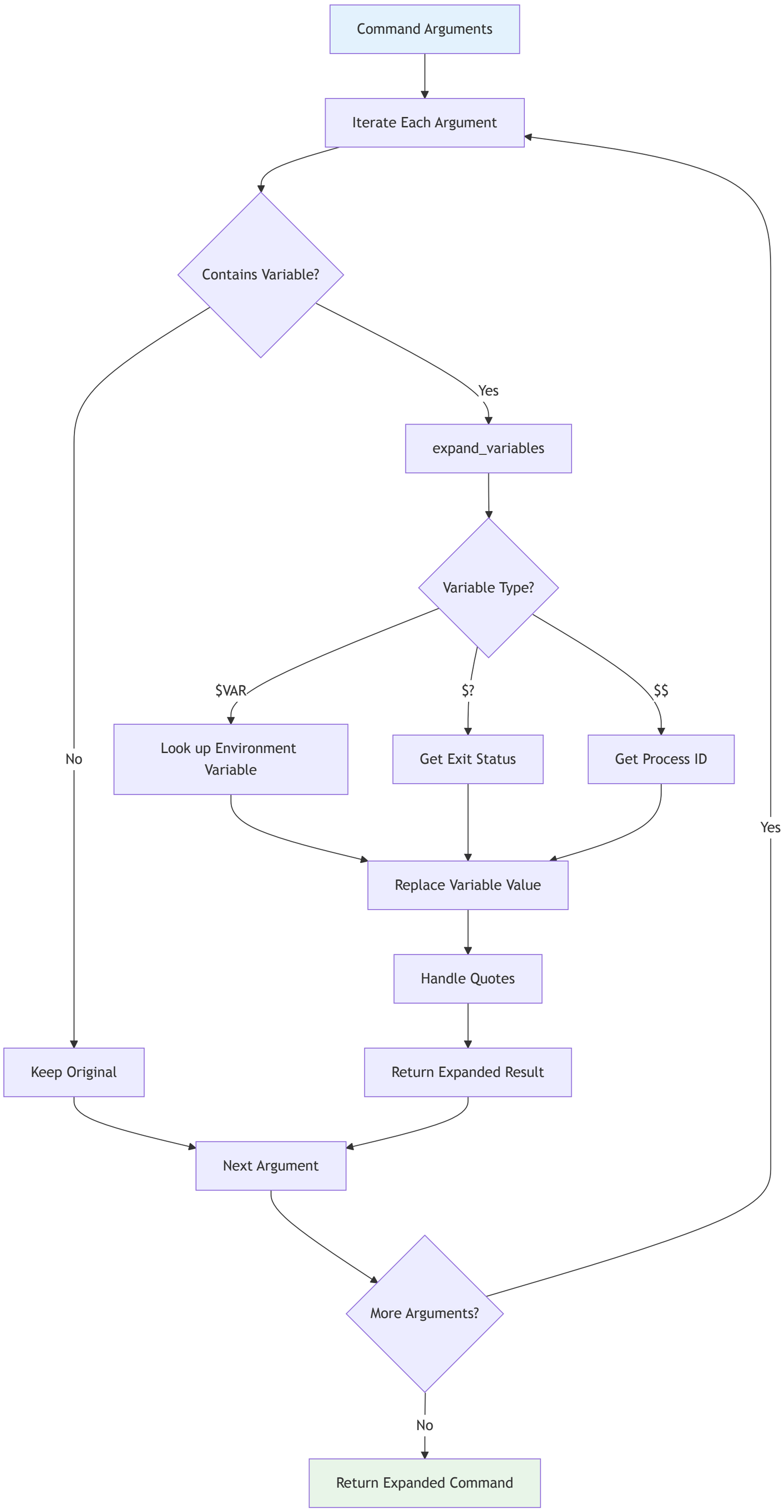
1. Lexer Flow



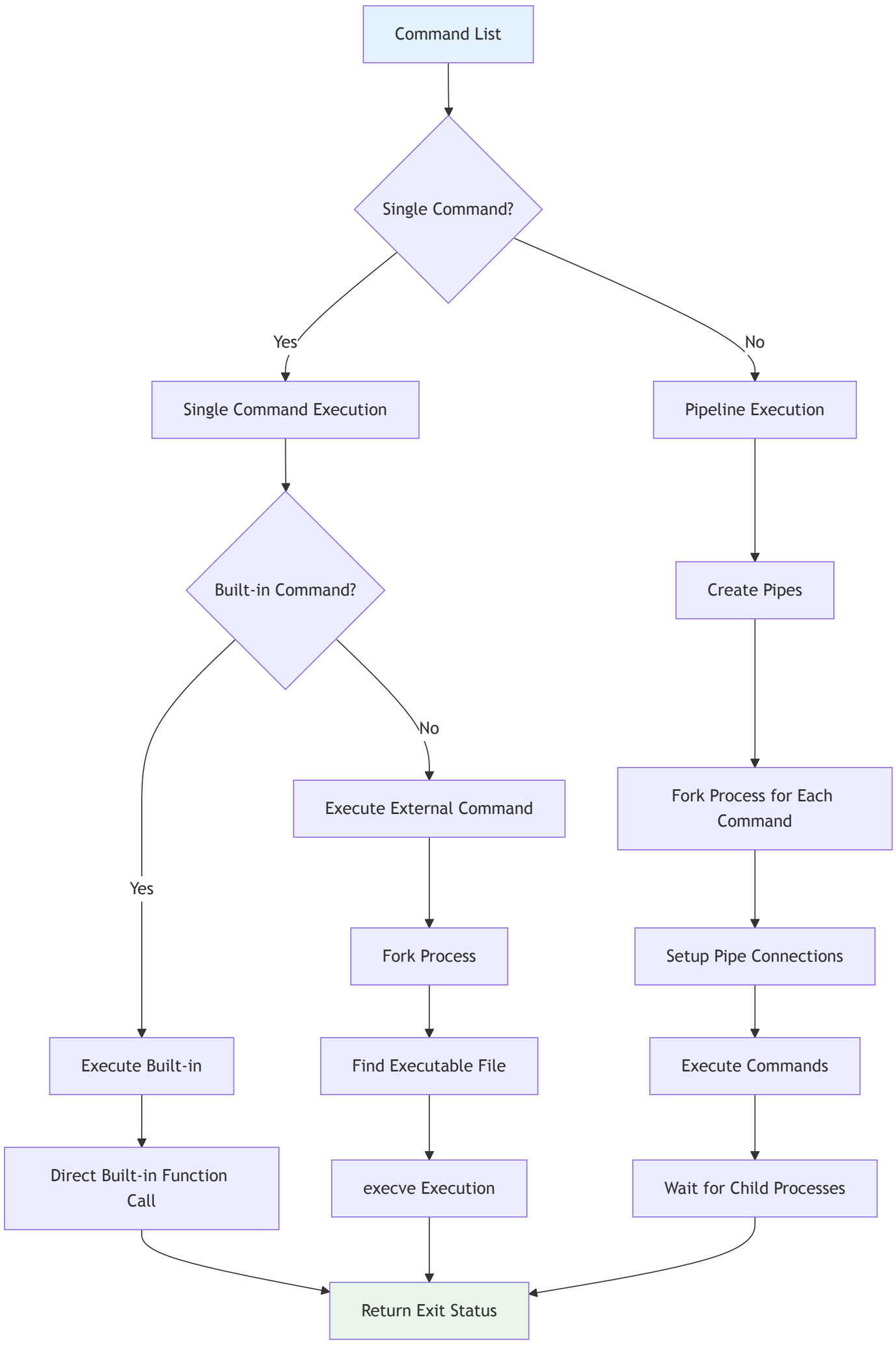
2. Parser Flow



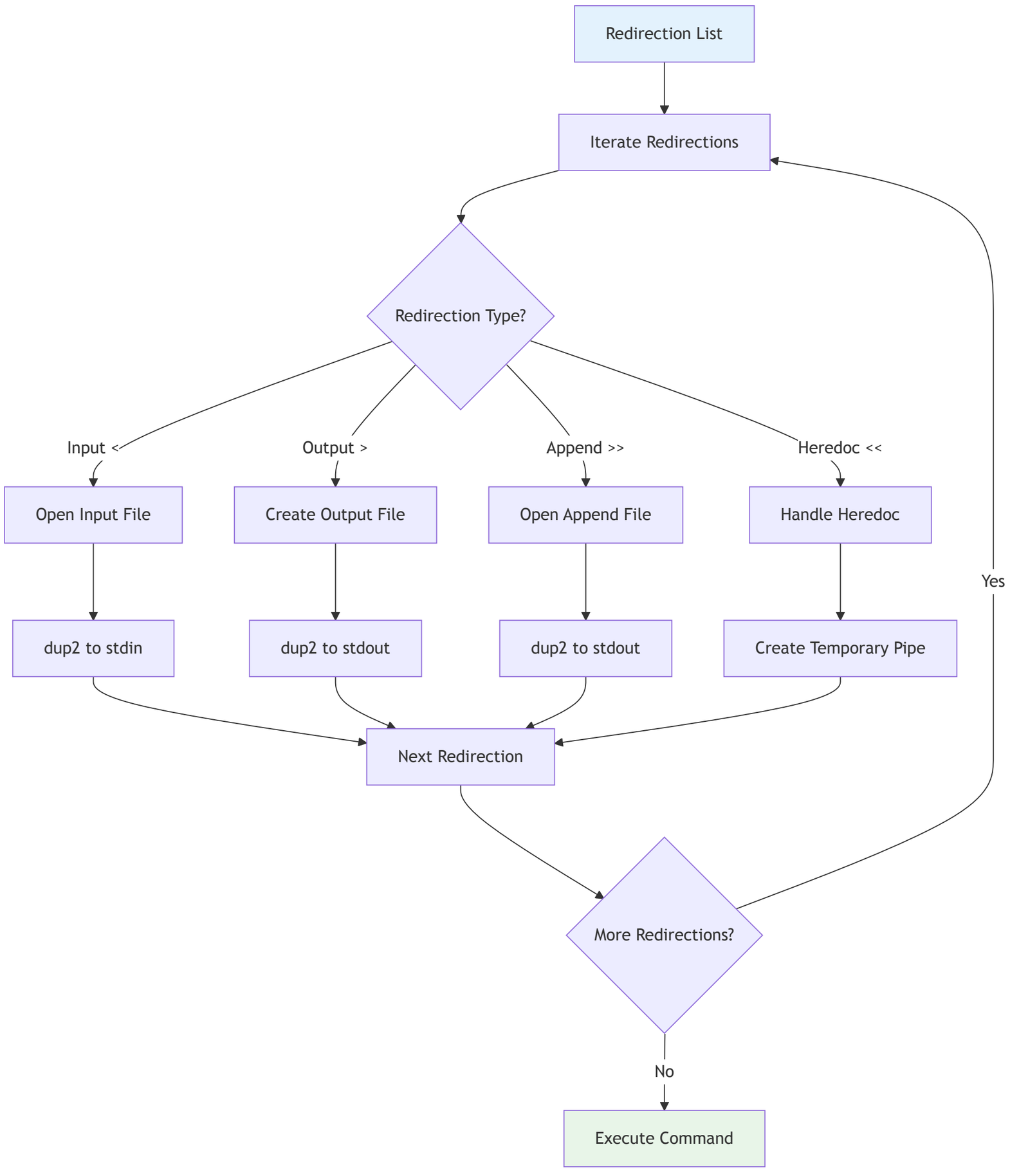
3. Variable Expansion Flow



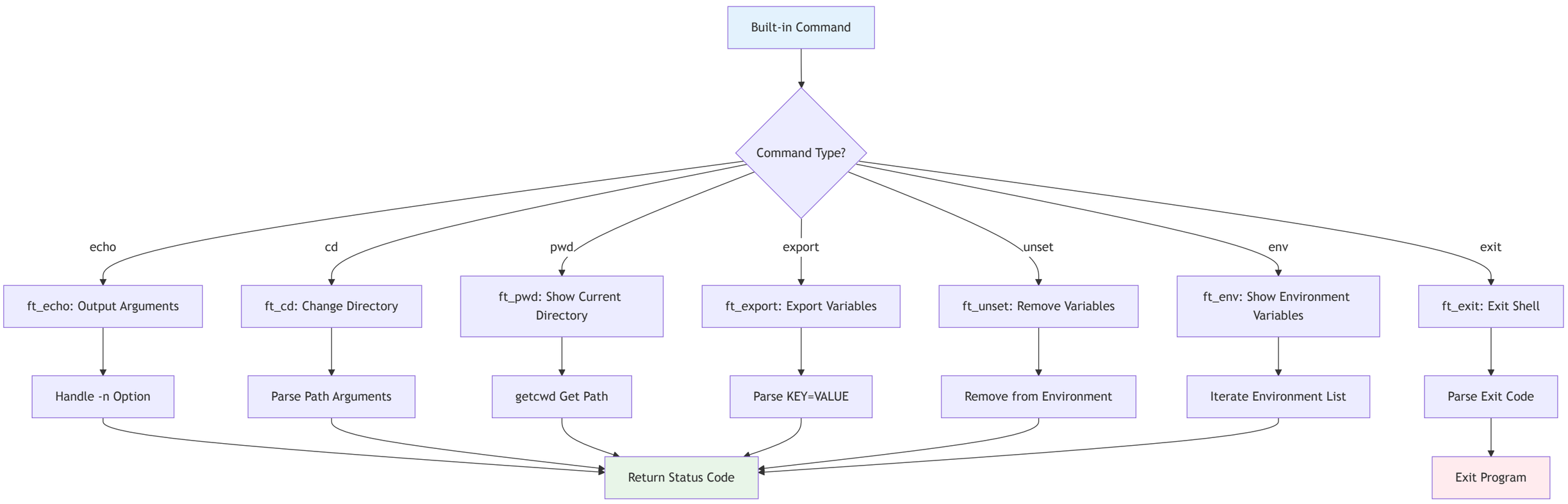
4. Executor Flow



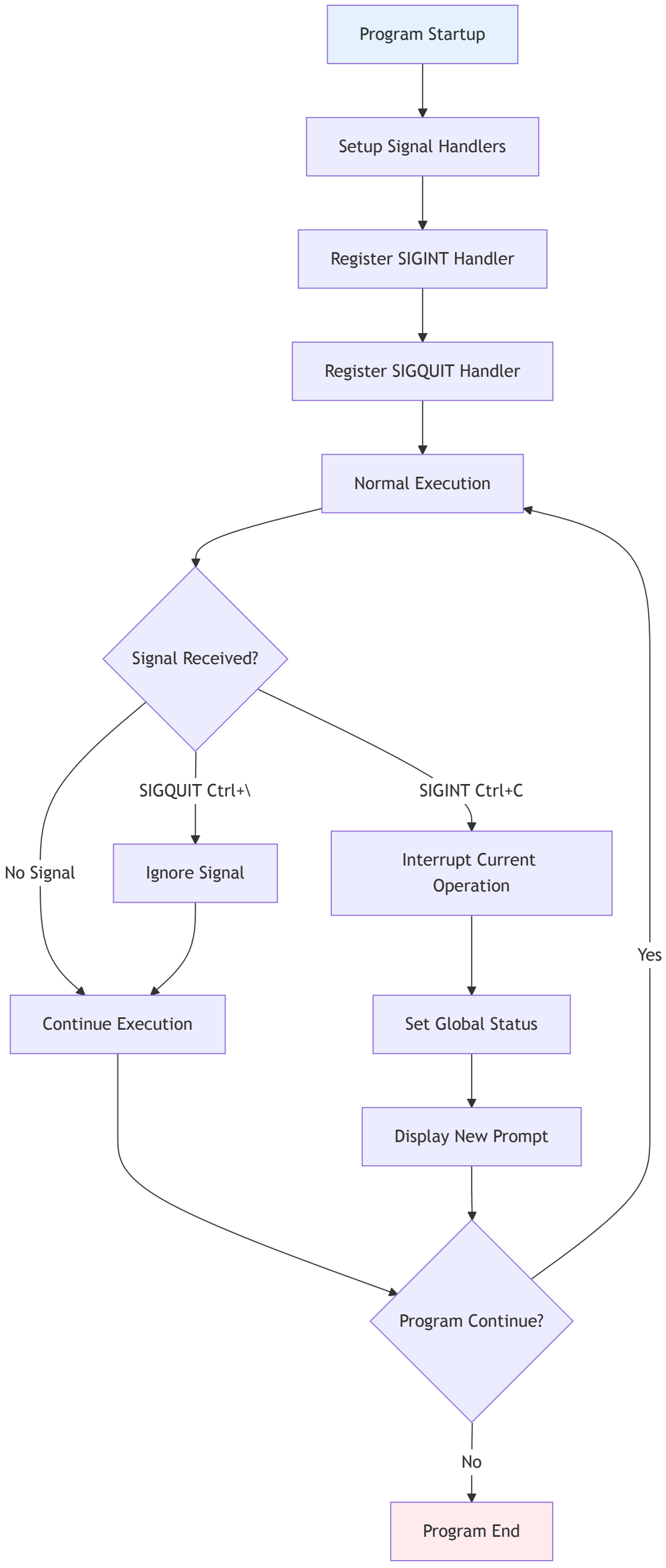
5. Redirection Handling Flow



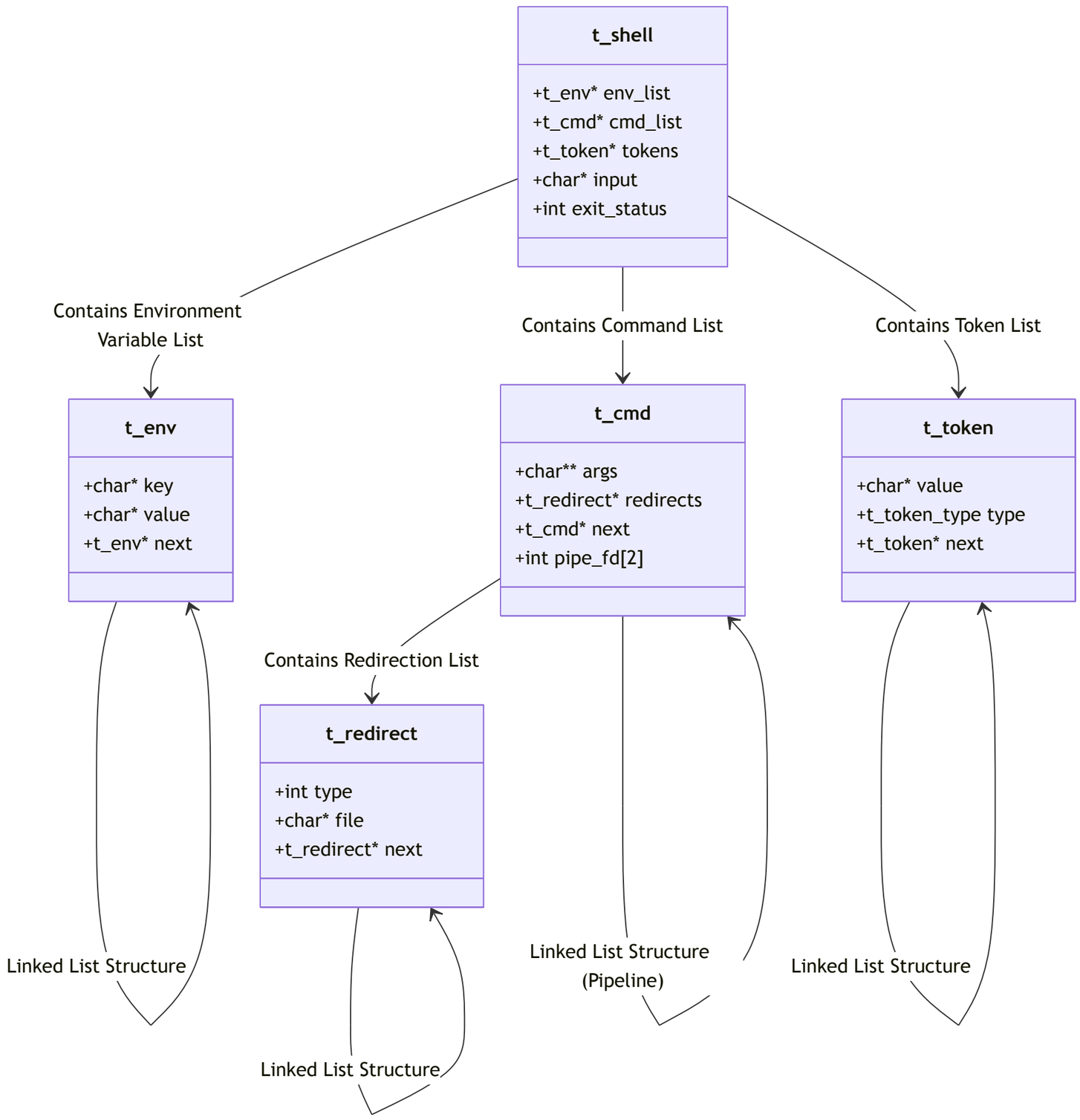
6. Built-in Commands Flow



7. Signal Handling Flow



Data Structure Relationship Diagram



File Organization Structure

```
minishell/
├── includes/
│   ├── minishell.h          # Main header file
│   └── executor.h           # Executor header file
├── srcs/
│   ├── main/               # Main program module
│   │   ├── main.c          # Program entry point
│   │   ├── input_handler.c # Input handling
│   │   ├── command_processor.c # Command processing
│   │   └── main_utils.c    # Main program utilities
│   ├── lexer/              # Lexical analyzer
│   │   ├── lexer.c         # Main lexical analysis
│   │   ├── lexer_word.c    # Word handling
│   │   ├── lexer_special.c # Special character handling
│   │   └── lexer_utils.c   # Lexical utilities
│   ├── parser/             # Syntax analyzer
│   │   ├── parser.c        # Main syntax analysis
│   │   ├── parser_args.c   # Argument parsing
│   │   ├── parser_redirect.c # Redirection parsing
│   │   └── parser_utils.c  # Parser utilities
│   ├── expansion/          # Variable expansion
│   ├── executor/           # Executor
│   │   ├── executor.c      # Main executor
│   │   ├── builtin_handler.c # Built-in command handling
│   │   ├── external_cmd.c  # External command handling
│   │   ├── pipe_handler.c  # Pipeline handling
│   │   ├── redirection_handler.c # Redirection handling
│   │   └── heredoc.c       # Heredoc handling
│   ├── builtins/           # Built-in commands
│   │   ├── cd.c            # cd command
│   │   ├── echo.c          # echo command
│   │   ├── env.c           # env command
│   │   ├── exit.c          # exit command
│   │   ├── export.c        # export command
│   │   ├── pwd.c           # pwd command
│   │   └── unset.c         # unset command
│   ├── env/                # Environment variable management
│   ├── signals/            # Signal handling
│   └── utils/              # General utilities
└── libft/                  # Custom library functions
```

Execution Flow Summary

1. **Program Startup**: Initialize shell structure, signal handling, readline
2. **Input Loop**: Continuously read user input

3. **Lexical Analysis**: Break input string into tokens
4. **Syntax Analysis**: Organize tokens into command structures
5. **Variable Expansion**: Process environment variables and special variables
6. **Command Execution**: Choose execution method based on command type
7. **Resource Cleanup**: Free memory, prepare for next loop iteration

Each module is independent and collaborates through well-defined interfaces, forming a complete shell interpreter system.