# Systems Programming in C++

CS330

**Nathan Warner**


Northern Illinois
University

Computer Science
Northern Illinois University
March 29, 2024
United States

## Contents

# C Library Functions

## 1.1 cstdlib utils

## 1.2 getenv

**Concept 1:** The **getenv** function is a standard library function that provides access to the environment variables of the process. Environment variables are dynamic-named values that affect the processes running on a computer. They can be used to configure system settings, pass configuration data to applications, and enable communication between different parts of an operating system or between different programs.

### Signature

```
char* getenv(const char* name);
```

Where name is a a C-string (const char*) representing the name of the environment variable whose value is being requested.

### Return Value

If the environment variable is found, getenv returns a pointer to a C-string containing the value of the variable.

If the environment variable is not found, it returns a null pointer (NULL in C, nullptr in C++).

### Example

```
#include <cstdlib>
#include <iostream>

int main() {
    // Attempt to retrieve the PATH environment variable
    const char* path = getenv("PATH");
    if (path != nullptr) {
        std::cout << "PATH: " << path << std::endl;
    } else {
        std::cout << "PATH environment variable not found." << std::endl;
    }
    return 0;
}
```

### 1.2.1 exit

**Signature**

```
void exit(int status)
```

The exit function is quite simple, it terminates the calling process

Zero for a successful termination, anything else is unsuccessful termination.

### 1.2.2 system

**Signature**

```
int system(const char* command)
```

The system command allows us to run shell commands. It invokes the command processor to execute a comamnd. The function returns the exit status of the command.

**Note:** If command is a nullptr, the function only checks if a command processor is available.

**Example**

```
int main(int argc, const char* argv[]) {
    int rs;
    if (!system(NULL)) {
        exit(EXIT_FAILURE);
    }
    cout << system("ls -la");

    return EXIT_SUCCSESS;
}
```

# Regular expressions library <regex.h>

## 2.1