



# Problem Solving With C - UE24CS151B

## Environment Variables

---

**Prof. Sindhu R Pai**

PSWC Theory Anchor, Feb-May, 2025

Department of Computer Science and Engineering

## PROBLEM SOLVING WITH C

### Environment Variables

---



- Introduction
- Standard Environment Variables
- Environment Access
- Demo of C code

## PROBLEM SOLVING WITH C

### Environment Variables

---



### Introduction

- A set of dynamic-named values that can affect the way the running processes behave on a computer
- They are part of the environment in which the process runs
- When a program is executed, it receives **information about the context in which it was invoked** using two ways -> **CLA and Environment variables**

```
int main(int argc,char*argv[],char*envp[])
```

- Programs executed from the shell inherit all of the environment variables from the shell

## PROBLEM SOLVING WITH C

### Environment Variables

---



#### Standard Environment Variables

- Used for information about the **user's home directory, terminal type, current locale** and etc.
- The set of all environment variables that have values is collectively known as the ***environment***
- Names of environment variables are **case - sensitive** and must contain the character '='
- The values of environment variables can be anything that can be represented as a string. **Must not contain an embedded null character**

## PROBLEM SOLVING WITH C

### Environment Variables

---



#### Environment Access

- **char \* getenv (const char \*name)**
  - Returns a string that is the value of the environment variable name
  - If the environment variable name is not defined, the value is a NULL pointer
- **int putenv (char \*string)**
  - Adds or changes the value of the environment variable
  - String is of the form '**name=value**'. If name is not there in the environment, string added to the environment. If name exist, name is updated with the new value
  - Returns 0 on success and else Non-zero and errno is set

## PROBLEM SOLVING WITH C

### Environment Variables

---



#### Environment Access continued..

- **int setenv (const char \*name, const char \*value, int replace)**
  - Used to add a new definition to the environment
  - On success, return 0. Else, environment is unchanged with -1 return value and errno is set
- **int unsetenv (const char \*name)**
- **int clearenv (void)**
- **char \*\* environ**
  - Represented as an array of strings. Each string is of the format '**name=value**'
- All these functions are available in **stdlib.h**

## PROBLEM SOLVING WITH C

### Environment Variables

---



#### Demo of C Code

- **Code to display the Environment variable and setting the Environment variable**



**THANK YOU**

---

Department of Computer Science and Engineering

Dr. Shylaja S S, Director, CCBD & CDSAML, PESU

Prof. Sindhu R Pai - [sindhurpai@pes.edu](mailto:sindhurpai@pes.edu)