UNIT -V

Dynamic Memory Allocation, Constructor , Destructor and Exploration of Header files

Header Files

- A header file in C++ contains:
  - Function definitions
  - Data type definitions
- Header files offer these features by importing them into your program with the help of a preprocessor directive called #include. These preprocessor directives are responsible for instructing the C++ compiler that these files need to be processed before compilation.
- C++ program should necessarily contain the header file <iostream> which stands for input and output stream used to take input with the help of “cin>>” function and display the output using “cout<<” function.

Header Files

- Basically, header files are of 2 types:
  - **Standard library header files:** These are the pre-existing header files already available in the C/C++ compiler.
  - **User-defined header files:** Header files starting #define can be designed by the user.

Header Files

- **User-defined header files:**
- **String:**
  - The library string.h (also referred as cstring) has several common functions for dealing with strings stored in arrays of characters. The string.h header file to be included before using any string function.
  - **1. strcpy()**
  - The strcpy() function takes two arguments: target and source. It copies the character string pointed by the source to the memory location pointed by the target. The null terminating character (\0) is also copied.

Header Files

- **strlen()**
- The strlen() takes a null terminated byte string source as its argument and returns its length. The length does not include the null(\0) character.
- **strcmp()**
- The strcmp() function takes two arguments: string1 and string2. It compares the contents of string1 and string2 lexicographically.
- The strcmp() function returns a:
- Positive value if the first differing character in string1 is greater than the corresponding character in string2. (ASCII values are compared)
- Negative value if the first differing character in string1 is less than the corresponding character in string2.
- 0 if string1 and string2 are equal.

Header Files

- **strcat()**
- The strcat() function takes two arguments: target and source. This function appends copy of the character string pointed by the source to the end of string pointed by the target.
- Any many more.....

Header Files

- **math.h:**
- Most of the mathematical functions are defined in math.h header file which includes basic mathematical functions.
- **cos() function**
- The cos() function takes a single argument in radians. The cos() function returns the value in the range of [-1, 1]. The returned value is either in double, float, or long double.
- **sqrt() function**
- The sqrt() function returns the square root of the given value of the argument. The sqrt() function takes a single non-negative argument. If a negative value is passed as an argument to sqrt() function, a domain error occurs.

Header Files

- **sin() function**
- The sin() function takes a single argument in radians. The sin() function returns the value in the range of [-1, 1]. The returned value is either in double, float, or long double.
- **pow() function**
- The pow() function returns base raised to the power of exponent. If any argument passed to pow() is long double, the return type is promoted to long double. If not, the return type is double. The pow() function takes two arguments:
- base - the base value
- exponent - exponent of the base

Header Files

- **cctype function**
- The C++ <cctype> header file declares a set of functions to classify (and transform) individual characters. For example, isupper() checks whether a character is uppercase or not.
- C++ isalpha()
  - checks if given character is alphabet or not
- C++ isblank()
  - checks if given character is a blank character
- C++ iscntrl()
  - checks if given character is control character
- C++ isdigit()
  - Checks if given character is a digit or not

Header Files

- **cctype function**
- ++ ispunct()
  - check if given character is punctuation character
- C++ isspace()
  - check if given character is whitespace character
- C++ isupper()
  - check if given character is uppercase or not
- C++ tolower()
  - Converts a given character to lowercase
- C++ toupper()
  - Converts a given character to uppercase

Header Files

- **Random**
  - The function void srand(unsigned int seed) seeds the random number generator used by the function rand.
  - The declaration of srand() is like below:
    - void srand(unsigned int seed)
  - It takes a parameter called seed. This is an integer value to be used as seed by the pseudo-random number generator algorithm. This function returns nothing.
  - To get the number we need the rand() method. To get the number in range 0 to max, we are using modulus operator to get the remainder.
  - For the seed value we are providing the time(0) function result into the srand() function.

Header Files

```
#include<iostream>
#include<cstdlib>
using namespace std;
main() {
   int max;
   max = 100; //set the upper bound to generate the random number
   srand(time(0));
   cout << "The random number is: "<<rand()%max;
}
```

Header Files

- **Create your own Header File**
  - Instead of writing a large and complex code, you can create your own header files and include it in the C/C++ library to use it whenever you wish as frequently as you like. It enhances code functionality and readability.
  - **Step-1:** Write your own code in C++ and save the file with a .h extension instead of a .cpp
  - The name of the file you save with .h extension would be the name of your header file.
  - **Step-2:** #include“headerfilename.h” – Enclosing the header file name within double quotes signifies that the header file of C++ is located in the present folder you are working with.
  - **Step-3:** After the code is written using your file with the .h extension, compile and run your program.

Header Files

- ***Important Note: The header file and C++ program should be in the same folder.***