

DAILY ASSESSMENT FORMAT

Date:	23-06-2020	Name:	Rajeshwari Gadagi
Course:	C++ Programming	USN:	4AL17EC076
Topic:	Module 3	Semester & Section:	6thSEM & 'B' SEC
Github Repository:	Rajeshwari-gadagi		

FORENOON SESSION DETAILS

Image of session

Data Types

The operating system allocates memory and selects what will be stored in the reserved memory based on the variable's **data type**. The data type defines the proper use of an identifier, what kind of data can be stored, and which types of operations can be performed.

There are a number of built-in types in C++. Tap Continue to learn more!

132 COMMENTS

→

int, float, double

Integers

The **integer** type holds non-fractional numbers, which can be positive or negative. Examples of integers would include 42, -42, and similar numbers.

The size of the **integer** type varies according to the architecture of the system on which the program runs, although 4 bytes is the minimum size in most modern system architectures.

173 COMMENTS

→

Data Types, Arrays, Pointers
string, char, bool

XP 127 1/3

Strings

A **string** is an ordered sequence of characters, enclosed in **double quotation marks**. It is part of the Standard Library. You need to include the `<string>` library to use the **string** data type. Alternatively, you can use a library that includes the **string** library.

```
#include <string>
using namespace std;

int main() {
    string a = "I am learning C++";
    return 0;
}
```

Try It Yourself

The `<string>` library is included in the `<iostream>` library, so you don't need to include `<string>` separately, if you already use `<iostream>`.

Data Types, Arrays, Pointers
Variable Naming Rules

XP 132 1/3

Variable Naming Rules

Use the following rules when naming variables:

- All variable names must begin with a letter of the alphabet or an underscore(_).
- After the initial letter, variable names can contain additional letters, as well as numbers. Blank spaces or special characters are not allowed in variable names.

There are two known naming conventions:
Pascal case: The first letter in the identifier and the first letter of each subsequent concatenated word are capitalized. For example: **BackColor**
Camel case: The first letter of an identifier is lowercase and the first letter of each subsequent concatenated word is capitalized. For example: **backColor**

255 COMMENTS

3077 Q&A →

Report – Report can be typed or hand written for up to two pages.

C++ Programming

Monday, 30th Jan 2020

→ Modules: Data Types, Storage, Pointers

→ Data Types

The C++ allocates memory & selects addressed size be decided while the measured memory decided on the individual data types.

* Expression

- Numeric Data Types

Numeric DT include Integers (whole no.) & Floating point no.

* Storage for Characters

A string is composed by no., characters, & symbols. String elements are placed in double quoted marks, some examples are "Hello", "My name is David", etc.

* Booleans

The Boolean data type returns first true (possible values: true() & false())

→ int, float, double

* The integer → adds non-decimal no., which can be positive or negative.

With the int keyword to define its integer data type, int a = 42;

decimal, unsigned, short, long

before +ve -ve otherwise the default
of datatype size 32b.

* floating point numbers

A floating point type variable can hold a real no. "float"

There are three different floating point data types: float, double & long double.

→ String, char, bool

* String

A string is an ordered sequence of characters, enclosed in double quotation marks.

* Character

char data variable holds a 1-byte storage.

A character is enclosed within single quotes (such as 'a', 'b', etc).

* Boolean

Boolean variables only have two possible values: true() & false()

→ Variables Naming Rules

All other variable names must begin with a letter of the alphabet or an underscore (-).

After the initial letter, variable names can contain additional letters, as well as numbers.

* Case-Sensitivity

C++ is case-sensitive, which means that can identify whether whitespace is not equivalent to identify one with underline in lowercase.

C++ keyword (reserved word) cannot be used as variable names. For ex: int, float, double, const

→ Array

An array is used to store a collection of data, and it may be useful to think of arrays as collections of variables that are all of the same type.

* Initializing array

If you omit the size of the array, an array will be created that can hold the whole section required.

* Accessing array elements

Index numbers can be used to assign a new value to an element.

> Using Arrays in Loops

> Arrays with Calculations

> Multidimensional arrays

It holds one or more arrays. Define a multidimensional array as follows:

type name [size][size]...[size];

> Introduction to Pointers

* Every variable is a memory location, which has its address defined.

* A pointer is a variable, with the address of another variable as its value.

* All pointers share the same data type - reading the address or that specific memory.

papergrid

Date: / /

There are two operators for pointers -

address - & operator ()

Content - * (dereference) operator (*)

> Static & Dynamic Memory

The static's all your local variable stays in memory from the start.

The heap is used program memory that comes in use when the program runs to dynamically allocate the memory.

> The sizeof() Operator

Size of

* While the size allocated for memory to data type depends on the architecture of compiler you use to run your program.

Date:	23-06-2020	Name:	Rajeshwari Gadagi
Course:	C++ Programming	USN:	4AL17EC076
Topic:	Module 4	Semester & Section:	6th SEM & 'B' SEC
Github Repository:	Rajeshwari-gadagi		

AFTERNOON SESSION DETAILS

Image of session

Functions

A function is a group of statements that perform a particular task. You may define your own functions in C++.

Using functions can have many advantages, including the following:

- You can reuse the code within a function.
- You can easily test individual functions.
- If it's necessary to make any code modifications, you can make modifications within a single function, without altering the program structure.
- You can use the same function for different inputs.

Every valid C++ program has at least one function - the `main()` function.

The Return Type

The `main` function takes the following general form:

```
int main()
{
    // some code
    return 0;
}
```

A function's **return type** is declared before its name. In the example above, the return type is `int`, which indicates that the function returns an `integer` value. Occasionally, a function will perform the desired operations without returning a value. Such functions are defined with the keyword `void`.

`void` is a basic data type that defines a valueless state.

Functions
Introduction to Functions

XP 162 4/6

Defining a Function

As an example, let's define a function that does not return a value, and just prints a line of text to the screen.

```
void printSomething()
{
    cout << "Hi there!";
}
```

Our function, entitled `printSomething`, returns `void`, and has no parameters. Now, we can use our function in `main()`.

```
int main()
{
    printSomething();
    return 0;
}
```

Try It Yourself

Functions
Function Parameters

XP 167 1/3

Function Parameters

For a function to use `arguments`, it must declare formal `parameters`, which are variables that accept the `argument`'s values.

For example:

```
void printSomething(int x)
{
    cout << x;
}
```

This defines a function that takes one `integer` parameter and prints its value.

Formal parameters behave within the function similarly to other local variables. They are created upon entering the function, and are destroyed upon exiting the function.

178 COMMENTS

Report – Report can be typed or hand written for up to two pages.

C++ Programming

23/06/2020 - Tuesday

Functions:

Introduction to Functions:

A function is a group of statements that performs a particular task.

The Return type:

A function's return type is declared before its name.

Defining a Function:

return-type function-name (parameters, body) of the function.

A function declaration, or function prototype, tells the compiler about a function so that it can call the function. (The actual body of the function can be defined separately.)

Function Parameters:

For a function to use arguments, it must declare formal parameters, which are variables that accept the arguments' values.

Once parameters have been declared, you can pass the corresponding argument when the function is called.

Function with multiple parameters:

- * Datatype of nouns should be informed for each parameter.
- * You can add as many parameters to a single function as you want.

The rand() function:

In the C++ Standard Library, you can access a female random number generator function that's called rand(). When used, it is required to include the header <cstdlib.h>

A for loop can be used to generate multiple random numbers.

* Using the <i> operator to generate random no. is another way to do it.

* The srand() function is used to generate truly random no.'s.

This function requires to specify a seed value as its parameter, which is used for the rand() function's algorithm.

* A function to generate truly random numbers is to use the <random.h> library to seed values for the srand() function.

> Default Value of Parameters

when defining a function, you can specify a default value for each of the parameters. If the corresponding argument is omitted when you call a function, it uses the default value.

> Function Overloading

function overloading allows to create many functions with the same name, so long as they have different parameters.

> Recursion

if recursive function meets all a function that calls itself.

> Passing arrays to Functions

arrays can also be passed to a function as arguments.

The parameter should be declared when array using square brackets, when declaring the function.

> Passing References with Pointers

there are two ways to pass arguments to a function is being called.

Copy value: this method copies the arguments' initial values into the function's formal parameters.

By reference: this method copies the arguments' reference into the formal parameters.