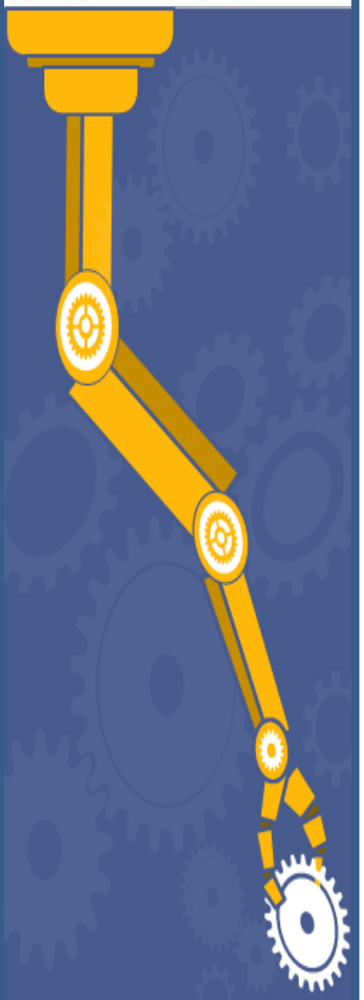
VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)



BCSE102L- Structured and object-oriented programming

Dr. P.Keerthika
Associate Professor
School of Computer Science & Engineering
VIT, Vellore.

BCSE102L- Structured and object-oriented programming

Module:1	**C Programming Fundamentals**	**2 hours**
Variables - Reserved words – Data Types – Operators – Operator Precedence - Expressions - Type Conversions - I/O statements - Branching and Looping: if, if-else, nested if, if-else ladder, switch statement, goto statement - Loops: for, while and do…while – break and continue statements.		
Module:2	**Arrays and Functions**	**4 hours**
Arrays: One Dimensional array - Two-Dimensional Array – Strings and its operations. User Defined Functions: Declaration – Definition – call by value and call by reference - Types of Functions - Recursive functions - Storage Classes - Scope, Visibility and Lifetime of Variables.		
Module:3	**Pointers**	**4 hours**
Declaration and Access of Pointer Variables, Pointer arithmetic – Dynamic memory allocation – Pointers and arrays - Pointers and functions.		
Module:4	**Structure and Union**	**2 hours**
Declaration, Initialization, Access of Structure Variables - Arrays of Structure - Arrays within Structure - Structure within Structures - Structures and Functions – Pointers to Structure -		
Module:5	**Overview of Object-Oriented Programming**	**5 hours**
Features of OOP - Classes and Objects - "this" pointer - Constructors and Destructors - Static Data Members, Static Member Functions and Objects - Inline Functions – Call by reference - Functions with default Arguments - Functions with Objects as Arguments - Friend Functions and Friend Classes.		
Module:6	**Inheritance**	**5 hours**
Inheritance - Types of Inheritance: Single inheritance, Multiple Inheritance, Multi-level Inheritance, Hierarchical Inheritance - Multipath Inheritance - Inheritance and constructors.		
Module:7	**Polymorphism**	**4 hours**
Function Overloading - Operator Overloading – Dynamic Polymorphism - Virtual Functions - Pure virtual Functions - Abstract Classes.		
Module:8	**Generic Programming**	**4 hours**
Function templates and class templates, Standard Template Library.		
	Total Lecture hours:	**30 hours**

BCSE102L- Structured and object-oriented programming – Text Books and Reference Books

Text Book(s)	
1.	Herbert Schildt, C: The Complete Reference, 4th Edition, McGraw Hill Education, 2017
2.	Herbert Schildt, C++: The Complete Reference, 4th Edition, McGraw Hill Education, 2017.
Reference Books	
1.	Yashavant Kanetkar, Let Us C: 17th Edition, BPB Publicaitons, 2020.
2.	Stanley Lippman and Josee Lajoie, C++ Primer, 5th Edition, Addison-Wesley publishers, 2012.

VIT
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

BCSE102P- Structured and object-oriented programming Laboratory

	Indicative Experiments
1.	Programs using basic control structures, branching and looping
2.	Experiment the use of 1-D, 2-D arrays and strings and Functions
3.	Demonstrate the application of pointers
4.	Experiment structures and unions
5.	Programs on basic Object-Oriented Programming constructs.
6.	Demonstrate various categories of inheritance
7.	Program to apply kinds of polymorphism.
8.	Develop generic templates and Standard Template Libraries.

Text Book(s)	
1.	Robert C. Seacord, Effective C: An Introduction to Professional C Programming, 1st Edition, No Starch Press, 2020.
Reference Book(s)	
1.	Vardan Grigoryan and Shunguang Wu, Expert C++: Become a proficient programmer by learning coding best practices with C++17 and C++20's latest features, 1st Edition, Packt Publishing Limited, 2020.

BCSE102L- Structured and Object-Oriented Programming

- **Module-6: Inheritance**
  - **Inheritance- Introduction & Types**
  - **Single Inheritance**
  - **Multiple Inheritance**
  - **Multilevel Inheritance**
  - **Hierarchical Inheritance**
  - **Multipath/Hybrid Inheritance**
  - **Inheritance and Constructors**

Inheritance and Constructors

- When constructors are present both in base and derived classes then how they are called, how values are passed from derived class to base class?

Inheritance& Constructor Example-1

```
class first
{
public :
first( )
{
cout<<"Hello from constructor of first"<<endl;
}
};
```

```
class second : first
{
public :
second( )
{
cout<<"Hello from constructor of second"<<endl;
}
};
```

Constructor in Single inheritance

```
int main( )
{
second obj;
}
```

OUTPUT:

Hello from constructor of first

Hello from constructor of second

Example-11

```
class human
{
public :
human( )
{
cout<<"Human constructor created"<<endl;
}
~ human()
{
cout<<"Human constructor destroyed"<<endl;
}
};
```

```
class men: public human
{
public :
men( ) : human() //
{
cout<<"men constructor created"<<endl;
}
~ men( )
{
cout<<"men constructor destroyed "<<endl;
}

};
```

Example-11

```
int main( )
{
men obj;
return 0;
}
```

OUTPUT: ??

```
Human constructor created
men constructor created
men constructor destroyed
Human constructor destroyed
```

Example-III

```
class first
{
int first_data;
public :
first (int x)
{
first_data = x;
cout<<"Con of first called";
}
void first_show( )
{
cout<<"FIRSTDATA
="<<first_data<<endl;
}
};
```

```
class second : public first
{
int second_data;
public :
second(int a, int b) : first(a)
{
second_data=b;
cout<<endl<<"Con of second
called"<<endl;
}
void second_show( )
{
first_show( );
cout<<"SECONDDATA="<<seco
nd_data<<endl;
}
};
```

Example-III

```
int main( )
{
second s(10,20);
s.second_show( );
}
```

OUTPUT: ??

```
Con of first called
Con of second called
FIRSTDATA =10
SECONDDATA=20
```