VAISHNO COLLEGE OF ENGINEERING

Affliated to HPTU, Hamirpur and approved by AICTE

Computer programming lab

Lab Manual

CSPC-111P (NEP Syllabus)

Department of Computer Science & Engineering

VillThapkour, PO Bhardoya, Tehsil Indora,Distt. Kangra (HP)-176403

Contact: 094183-18394, Web: www.vaishno.edu.in

Vision of Institute

To emerge as an institute of eminence in the fields of engineering, technology and management in serving the industry and the nation by empowering students with a high degree of technical managerial and practical competence.

Mission of Institute

M1 To strengthen the theoretical, practical and ethical dimensions of the learning process by fostering a cultural of research and innovation among faculty members and students.

M2 To encourage long term interaction between academia and industry through the involvement of industry for hands on implementation of the curriculum.

M3 To strengthen and molding students in professional ethical, social and environmental dimensions by encouraging participation in co-curricular extracurricular and CSR activities.

Vision of the Department

To emerge as a department of eminence in computer science and engineering in serving the industry and the nation by empowering students with high degree of technical and practical competence.

Mission of the department

M1 To strengthen the theoretical and practical aspects of learning process by strongly encouraging a computer cultural of research, innovation and hands on learning in computer science and engineering

M2 To encourage long term interaction between the department and IT industry, through the involvement of IT industry for hands on implementation of course curriculum.

M3 To widen the awareness of students in professional, ethical, social and environmental dimensions by encouraging their participation in co-curricular extracurricular and CSR activities.

Program Educational Objectives (PEOs) of the department

PEO 1: Engage in successful careers in industry, academia, and public service, by applying the acquired knowledge of Science, Mathematics and Engineering, providing technical leadership for their business, profession and community

PEO 2: Establish themselves as entrepreneur, work in research and development organization and pursue higher education

PEO 3: Exhibit commitment and engage in lifelong learning for enhancing their professional and personal capabilities.

PROGRAM OUTCOMES

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: Problem analysis: Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able tocomprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one’s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO 12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcome (PSOs)

PSO1: Apply knowledge of mathematics, engineering sciences and multidisciplinary knowledge to the solution of computer science engineering problems.

PSO2: Apply research-based knowledge, appropriate techniques, IT tools to complex computer science engineering problems including design, analysis, interpretation of data, and synthesis of the information to provide valid conclusions.

PSO3: Apply ethical principles engineering profession and recognize the need of independent and lifelong learning for professional development and personnel growth.

Lab Syllabus & List of Experiments

CS-111P Computer Programming Lab							
Teaching Scheme			Credit	Marks Distribution			Duration End Semester Examination
L	T	P/D	C	Internal Assessment	End Semester Examination	Total	
0	0	2	1	Maximum Marks: 30	Maximum Marks: 20	50	2hrs
				Minimum Marks: 12	Minimum Marks: 08	20	

Following is the list of experiments/ jobs. Minimum 08 number of practicals are to be performed from following list. The additional experiments may be performed by the respective institution depending on the infrastructure available.

Laboratory Work:

1. WAP for basic input/output statement and various control statements.
2. WAP to create for function and function calling methods
3. WAP to take input and display elements of 1D and 2D array.
4. WAP for structures and display the values of structure members using structure variable.
5. WAP for creating class, defining member in class and accessing member.
6. WAP using various string functions in C++.
7. WAP for constructor and Destructor.
8. WAP for inheritance.
9. WAP for friend function and friend class.
10. WAP for polymorphism.
11. WAP for exception handling in C++.
12. WAP using template concept.

Evaluation Scheme

Internal Assessment: 30 marks (pass marks:12)

Distribution of marks for internal assessment:

- Written/presentation/Demonstration: 05
- Viva-voice: 05
- Teacher assessment: Lab Work performance/Report/File Work:15
- Attendance: 05

External Assessment: 20 marks (pass marks: 08)

Total marks 30+20=50, Pass marks = 20

Note: Student has to pass internal & external assessment separately.

GENERAL GUIDELINES AND SAFETY INSTRUCTIONS

1. You may use the computers in the lab only when a teacher is present.
2. Please place your bags at the front of the lab.
3. Do not eat or drink in the lab.
4. Keep the lab clean and neat at all times.
5. Use only the computer you are assigned to.
6. Report any hardware fault immediately to your teacher. Never attempt to dismantle the different parts of the computer.
7. Each student must log in to his/her account. No sharing of accounts is permitted.
8. The computers are for your academic use. Playing computer games for entertainment is strictly not allowed.
9. Shut down the computer properly after use.
10. Do not charge your personal mobile devices in the lab.

Cleanliness

- Keep your workspace clean and free of clutter
- Don't eat or drink in the lab
- Don't litter
- Don't remove cables or items from the lab

Fire safety

- Have a fire extinguisher and first-aid kit available
- Follow fire safety guidelines
- Be aware of the possibility of an accidental fire
- Know how to react to a fire
- Have a planned fire escape route

Eye and body safety

- Avoid eye fatigue by blinking often or closing your eyes for a few minutes
- Sit straight and in a comfortable posture
- Spread your fingers apart or rotate your wrists at regular intervals
- Wear proper lab attire
- Practice good hygiene

Other safety guidelines

- Don't spill liquids on the computer
- Don't touch hot or high voltage areas of printers

- Don't open a power supply or CRT monitor
- Don't tamper with wires or network cables
- Don't use illegal software
- Don't attempt to compromise network security

Practical No: 1

Aim:- WAP for basic input/output statement and various control statements.

Software Used:- Visual Studio Code

```
#include <iostream>

using namespace std;

int main()

{

int a, b;

 cout<<”enter two numbers:”<<endl;

cin>>a>>b;

if(a>b)

cout<<”the highest number is :”<<a;

else
cout<<” the highest number is :”<<b;

return 0;

}
```

Practical No: 2

Aim:- WAP to create for function and function calling methods

Software Used:- Visual Studio Code

```
# include <iostream>

using namespace std;

void swap(int &num1, int &num2)

{

    int temp;

    temp= num1;

    num1= num2;

    num2= temp;


}

int main()

{

    int a=5, b=4;

    cout<<"\n before swapping"<<"\n A="<<a<<"\n B="<<b<<endl;


    swap(a,b);

     cout<<"\n after swapping"<<"\n A="<<a<<"\n B="<<b<<endl;

    return 0;
}
```

Practical No: 3

Aim:- WAP to take input and display elements of 1D and 2D array.

Software Used:- Visual Studio Code

```
# include <iostream>
using namespace std;
int main()
{
   int marks[5] = {8,7,4,3,9};
   cout<<" the marks of students are:"<<endl;
   for(int i= 0; i<= 4; i++)
   cout<<" marks of "<<i+1  <<"students is"<<marks[i]<<endl;


}
```

Practical No: 4

Aim:- WAP for structures and display the values of structure members using structure variable.

Software Used:- Visual Studio Code

```
#include<iostream>
using namespace std;
struct Person
{
        int citizenship;  // struct members
        int age;
};              // end od sturcture body
int main(void) {
        struct Person p; // object create`
        p.citizenship = 1;
        p.age = 27;
        cout << "Person citizenship: " << p.citizenship << endl;
        cout << "Person age: " << p.age << endl;

        return 0;
}
```

Practical No: 5

Aim:- WAP for creating class, defining member in class and accessing member.

Software Used:- Visual Studio Code

```
 #include<iostream>
 using namespace std;
class student
{
  private:
  int enroll_no;
  char name[30];
  public:
```

```
 void getdata()

 {

   cout <<"enter enroll no. and name of student"<< endl;

 cin>>enroll_no>>name;


}


void displaydata()

{

 cout<<"Enrollment no"<< "\t"<< enroll_no<<endl;

 cout<<"Name is" << name<<endl;


}

};

int main()

{

 student s1;

 s1.getdata();

 s1.displaydata();

 return 0;


}
```

Practical No: 6

Aim:- WAP for constructor and Destructor.

Software Used:- Visual Studio Code

```
#include <iostream>
using namespace std;
class Hello
{
    public:
    Hello()    // constructor
    {
        cout<<" constructor function is called"<< endl;

    }
    ~ Hello()     // destructor
    {
        cout<<" dectructor function is called " << endl;

    }
    void display()   // member function
    {
        cout <<" hello world" << endl;

    }
};
int main ()
{
    Hello obj;

    obj.display();

    return 0;


}
```

Practical No: 7

Aim:- WAP for inheritance.

Software Used:- Visual Studio Code

```
#include <iostream>

using namespace std;


// base class

class Animal {


   public:

    void eat()

    {

        cout << "I can eat!" << endl;

    }


    void sleep()

    {

        cout << "I can sleep!" << endl;
    }
};

// derived class
class Dog : public Animal
{

   public:
    void bark()
    {
        cout << "I can bark! Woof woof!!" << endl;
    }
};
```

Practical No: 8

Aim:- WAP for exception handling in C++.

Software Used:- Visual Studio Code

```
#include <iostream>
using namespace std;
 int main()
 {
     int a,b,c;
     cout<< "Enter value of a:";
     cin>> a;
     cout<< " Enter value of b:";
```

```
    cin>> b;
    try
    {
        if (b!=0)
        {
            cout <<"Division ="<<c;

        }
        else
        {
            throw(b);
    }
    }
    catch (int b)
    {
        cout<<"division by <<b";
    }
}
```

Viva Questions

1. What do you mean by c++. & discuss the role of c++.
2. What do you mean by identifiers and keywords.
3. What do you mean by header files.

4. What do you mean by compilation.
5. Explain built-in datatypes.
6. What is the use of break and continue.
7. What is the use switch case.
8. When we use for loop .
9. What do you mean by function.
10. What do you mean by local & global variables & where we declare these variable in the code.
11. What do you mean by array, structure & class.
12. What do you mean by constructor & destructor.
13. How we declare friend function.
14. What do you mean by Inheritance .
15. Explain abstract class.
16. What do you mean by run time polymorphism.
17. How we declare virtual function.
18. Explain the role of templates in the code.
19. Explain catch & throw block.
20. How we handle with exception.

Laboratory Experiment Evaluation Rubric

Category	**Outstanding (Up to 100%)**	**Accomplished (Up to 75%)**	**Developing (Up to 50%)**	**Beginner (Up to 25%)**
Written/Presentation/Demonstration	The write-up is clear, well-organized, and follows the prescribed	The report follows the specified format, but some sections	The report includes most sections but lacks clarity, coherence, or	The report is poorly written and organized. Many

Category	Outstanding (Up to 100%)	Accomplished (Up to 75%)	Developing (Up to 50%)	Beginner (Up to 25%)
	format. All required sections (aim, apparatus, theory, procedure, diagram, etc.) are present and well-written. Demonstration is clear and thorough.	(like the diagram or theory) are missing or incomplete. The demonstration is understandable but lacks depth.	completeness in some parts (e.g., diagram missing, unclear theoretical explanation). The demonstration is incomplete or unclear.	sections are missing or incorrect (e.g., no diagram, incomplete procedure). The demonstration lacks clarity or is missing.
Viva-Voice	Demonstrates a deep understanding of the experiment, underlying principles, and outcomes. Answers questions confidently and accurately.	Demonstrates a general understanding of the experiment and principles but struggles with some aspects. Provides correct answers to most questions.	Struggles with some fundamental concepts and principles. Answering questions requires additional prompts, with a few errors in understanding.	Lacks a basic understanding of the experiment. Unable to answer most questions accurately. Demonstrates significant gaps in knowledge.

Category	**Outstanding (Up to 100%)**	**Accomplished (Up to 75%)**	**Developing (Up to 50%)**	**Beginner (Up to 25%)**
Performance/Report/File Work	Performs the experiment accurately and efficiently. The report is thorough, with correct observations, calculations, and analysis. Data is recorded neatly and with appropriate units. All relevant calculations and interpretations are included.	Performs the experiment well with minor errors or delays. The report is complete but may contain some inaccuracies or missing components in calculations or observations.	Completes the experiment but with notable mistakes, either in the setup or the data. The report has several missing or inaccurate components, including incorrect or incomplete calculations.	Struggles to perform the experiment correctly. Significant errors in setup, data collection, and analysis. The report is poorly structured with major inaccuracies or missing sections.
Attendance	Consistently attends all lab sessions, actively participates, and engages with the experiment and group discussions.	Attends most lab sessions with occasional absences. Participation is generally good but lacks consistency or depth.	Attends some lab sessions but has frequent absences or minimal participation.	Misses several lab sessions and shows minimal to no participation in class or group activities.