C++ MANUAL

PRESENTED BY

S.SINDHUJA

1. **Write a C++ Program to display names, roll no’s, and grades of 3 students who have appeared in the examination. Declare the class of name, roll no’s and grade. Create an array of class objects. Read and display the contents of the array.**

#include<iostream>

#include<conio.h>

using namespace std;

class stud

{

public:

char name[12];

int rollno;

char grade[2];

};

stud st[3];

int main()

{

int k=0;

while(k<3)

{

system("cls");

//gotoxy(2,4);

cout<<"Name : ";

//gotoxy(17,4);

cin>>st[k].name;

//gotoxy(2,5);

cout<<"Roll No. : ";

//gotoxy(17,5);

cin>>st[k].rollno;

//gotoxy(2,6);

cout<<"Grade: ";

//gotoxy(17,6);

cin>>st[k].grade;

st[k].grade[1]='\0';

puts("press any key...");

getch();

k++;

}

k=0;

//clrscr();

cout<<"\n Name\t Rollno \t Grade\n";

while(k<3)

{

cout<<st[k].name<<"\t"<<st[k].rollno<<"\t"<<st[k].grade<<"\n";

k++;

}

return 0;

}

1. **Write a C++ program to declare *struct*. Initialize and display contents of member variables.**

#include<iostream>

using namespace std;

struct item

{

int codeno;

float prize;

int qty;

};

int main()

{

item a,\*b;

system("cls");

a.codeno=123;

a.prize=150.75;

a.qty=150;

cout<<"\n With simple variable";

cout<<"\n Codeno:" <<a.codeno;

cout<<"\n Prize:"<<a.prize;

cout<<"\n Qty:"<<a.qty;

b=&a;

b->codeno=124;

b->prize=200.75;

b->qty=75;

cout<<"\n with pointer to structure";

cout<<"\n codeno:"<<b->codeno;

cout<<"\n prize"<<b->prize;

cout<<"\n qty:"<<b->qty;

}

1. **Write a C++ program to declare a class. Declare pointer to class. Initialize and display the contents of the class member.**

#include<iostream>

using namespace std;

class man

{

public:

char name[10];

int age;

};

int main()

{

man m={"RAVINDRA", 15};

man \*ptr;

ptr=&m;

system("cls");

cout<<"\n"<<m.name<<" " <<m.age;

cout<<"\n"<<ptr->name<<" "<<ptr->age;

}

**4) Given that an EMPLOYEE class contains following members:**

**data members: Employee number, Employee name, Basic, DA, IT, Net Salary and print data members.**

1. **Write a C++ program to read the data of N employee and compute Net salary of each employee (DA=52% of Basic and Income Tax (IT) =30% of the gross salary) .**

#include <iostream>

#include <conio.h>

using namespace std;

class employee

{

char name[10];

int no;

float basic;

float da;

float it;

float ns;

float gs;

public:

void input()

{

cout <<"Enter number:"; cin >> no;

cout <<"Enter name:"; cin >> name;

cout <<"Enter salary:"; cin >> basic;

}

void calculate()

{

da = 0.52 \* basic;

gs = da + basic;

it = 0.3 \* gs;

ns = gs - it;

}

void output()

{

cout<<no<<'\t'<<name<<'\t'<<basic<<'\t'<<ns<<'\t'<<gs <<'\n';

}

};

int main()

{

employee emp[20];

int n,i;

system("cls");

cout << "Enter no of employees:";

cin >> n;

for(i=0;i<n;i++)

{

emp[i].input();

emp[i].calculate();

}

cout<<"NUMBER"<<'\t'<<"NAME"<<'\t'<<"BASIC"<<'\t'<<"NET"<<'\t'<<"GROSS" << "\n";

for(i=0;i<n;i++)

{

emp[i].output();

}

getch();

}

1. **Write a C++ to illustrate the concepts of console I/O operations.**

#include<iostream>

#include<conio.h>

using namespace std;

int main()

{

cout.width(5);

cout<<"A";

cout.width(15);

cout<<"B";

cout.precision(2);

cout<<3.1452;

cout.fill('/');

cout.width(20);

cout<<"WEL"<<endl;

cout.fill('-');

cout.width(10);

cout<<"DONE";

}

**7) Write a C++ program to use scope resolution operator. Display**

**the various values of the same variables declared at different scope levels.**

#include<iostream>

#include<conio.h>

using namespace std;

int a = 10;

int main()

{

system("cls");

int a=20;

cout<<"::a="<<::a;

cout<<"a="<<a;

return 0;

}

1. **Write a C++ program to allocate memory using *new* operator.**

#include<iostream>

#include<conio.h>

using namespace std;

int main()

{

system("cls");

int \*p= new int[3],k;

for(k=0;k<3;k++)

{

cout<<"\n Enter a number: ";

cin>>\*p;

p++;

}

p-= 3;

cout<<"\n Entered numbers with their address are: \n";

for(k=0;k<3;k++)

{

cout<<"\n\t"<<\*p<<"\t"<<p;

p++;

}

p-= 3;

delete p;

}

**9) Write a C++ program to create multilevel inheritance. Create classes A1,A2, A3.**

#include<iostream>

#include<conio.h>

using namespace std;

class A1

{

protected:

char name[15];

int age;

};

class A2 : public A1

{

protected:

float height,weight;

};

class A3: public A2

{

protected :

char sex;

public:

void get()

{

cout<<"Name : ";

cin>>name;

cout<<"Age : ";

cin>>age;

cout<<"Sex : ";

cin>>sex;

cout<<"Height : ";

cin>>height;

cout<<"Weight : ";

cin>>weight;

}

void show()

{

cout<<"\nName : "<<name;

cout<<"\nAge : "<<age<<"Years";

cout<<"\nHeight : "<<height<<"Feets";

cout<<"\nSex : "<<sex;

cout<<"\n Weight : "<<weight<<"kg.";

}

};

int main()

{

System(“cls”);

A3 x;

x.get();

x.show();

getch();

}

**10) Write a C++ program to create an array of pointers. Invoke functions using array objects.**

#include<iostream>

#include<conio.h>

using namespace std;

class A

{

public:

virtual void show()

{

cout<<"A\n";

}

};

class B:public A

{

public:

void show()

{

cout<<"B\n";

}

};

class C:public A

{

public:

void show()

{

cout<<"C\n";

}

};

class D:public A

{

public:

void show()

{

cout<<"D\n";

}

};

class E:public A

{

public:

void show()

{

cout<<"E\n";

}

};

int main()

{

system("cls");

A a;

B b;

C c;

D d;

E e;

A \*pa[] = { &a,&b,&c,&d,&e};

for(int j=0;j<5;j++)

{

pa[j]->show();

getch();

}

}

**11) Write a C++ program to use pointer for both base and derived classes and call the member function. Use Virtual keyword.**

#include<iostream>

#include<conio.h>

using namespace std;

class super

{

public:

virtual void display()

{

cout<<"\n In function display() class super";

}

virtual void show()

{

cout<<"\n In function show() class super";

}

};

class sub: public super

{

public:

void display()

{

cout<<"\n In function display() class sub";

}

void show()

{

cout<<"\n In function show() class sub";

}

};

int main()

{

system("cls");

super S;

sub A;

super \*point;

cout<<"\n Pointer point points to class super\n";

point= &S;

point->display();

point->show();

cout<<"\n Now pointer point points to derived class sub\n";

point= &A;

point->display();

point->show();

return 0;

}