Solution

Fahaheel Al-Watanieh Indian Private School, Ahmadi, Kuwait

First Term Weekly Test #2 (15-May-2016)

Computer Science

Class 12

1. a) i) **Encapsulation:** Wrapping up of data members and associated member functions in a  
    single unit , called class, is known as encapsulation.

ii) Default visibility mode in a class is private whereas that in a structure is public.

b)

|  |  |
| --- | --- |
| **Constructor** | **Destructor** |
| A constructor executes whenever an object of a class is created. | A destructor is executed whenever the scope of an object gets over. |
| Constructor can be overloaded. | Destructor cannot be overloaded. |

Copy Constructor: voterid(voterid &); Destructor: ~voterid();

c)

|  |  |
| --- | --- |
| **Multiple Inheritance** | **Hierarchical Inheritance** |
| A class is derived from 2 or more base classes. | 2 or more classes are derived from a base class. |
| Example:  class A  {  int x;  public:  void f1();  };  class B  {  int y;  public:  void f2();  };  class C:public A, B  {  Int z;  public:  void f3();  }; | Example:  class A  {  int x;  public:  void f1();  };  class B:public A  {  int y;  public:  void f2();  };  class C:public A  {  Int z;  public:  void f3();  }; |

d) i) fstream df(“RECORDS.DAT”, ios::binary|ios::in);

ii) ifstream df; df.open(“RECORDS.DAT”, ios::binary|ios::in);

1. 2:60&60

5:75&70

2:55&60

1. #include <fstream.h>

void main()

{

**fstream** myfile(“HOLIDAY.DAT”,ios::in|ios::out);

**char ch;**

**while (myfile.**get**(ch)**)

**if (ch>=’a’ && ch<=’a’)class** Door;

{

ch-=32;

myfile.seekg(**-1, ios::cur**);

myfile<<ch;

}

**myfile.close();**

**}**

1. class SUPPLY

{

int code;

char foodname[20];

char sticker[20];

char foodtype[20];

void gettype()

{

if (strcmpi(sticker,"GREEN")==0)

strcpy(foodtype, "Vegetarian");

else if (strcmpi(sticker, "YELLOW")==0)

strcpy(foodtype, "Contains Egg");

else if (strcmpi(sticker, "RED")==0)

strcpy(foodtype, "Non-Vegetarian");

}

public:

void FoodIn()

{ cin>>code;

gets(foodname);

gets(sticker);

gettype();

}

void foodout()

{ cout<<code<<", "<<doodname<<", "

<<sticker<<", "<<foodtype<<endl;

}

};

1. legs, animaltype, animals, workers
2. pawsize, foodcost, jawht, cagesize, legs, animaltype, animals, workers
3. jawht, cagesize, datain(), dataout()
4. No
5. 84

1. a)

void Q6A()

{

ifstream f("PLACES.TXT);

char line[80];

int found = 0;

while (f.getline(line, 80))

{

if (line[0]=='P' || line[0]=='S')

{ cout<<line<<endl;

found = 1;

}

}

f.close();

if (found == 0)

cout<<"No line starting with P or S found in the file";

}

b)

void Q6B()

{

ifstream f("SCHOLAR.TXT");

char ch, word[20];

int count = 0;

while (f.get(ch))

cout<<ch;

f.close();

f.open("SCHOLAR.TXT");

while (f>>word)

if (word[0]!='O' && word[0]!='E')

count++;

f.close();

cout<<count;

}

c)

void Q6C()

{

ifstream f("MYTRIP.TXT");

char ch;

int count = 0;

while (f.get(ch))

{

cout<<ch;

if (isalpha(ch))

count++;

}

f.close();

cout<<count;

}

1. a)

void Q7a(int n)

{

ofstream df("laptop.dat",ios::app|ios::binary);

laptop L;

for (int i = 1; i<=n; i++)

{

L.newlaptop();

df.write((char\*)&L, sizeof(L));

}

int bytes = df.tellg();

int recs = bytes/sizeof(L);

df.close();

cout<<"File contains "<<recs<<" records";

}

b)

void Q7b()

{

ifstream df("laptop.dat",ios::binary);

laptop L;

int found = 0;

while(df.read((char\*)&L, sizeof(L)))

{ if (L.retram()==8 && L.retprice()<250)

{ L.laptopdisplay();

found = 1;

}

}

df.close();

if (found == 0)

cout<<"No such record found";

}

c)

void Q7c()

{

fstream df("laptop.dat",ios::binary|ios::in|ios::out);

laptop L; double pr;

while(df.read((char\*)&L, sizeof(L)))

{

pr = L.retprice();

pr = pr-0.2\*pr;

L.editprice(pr);

df.seekg(-sizeof(L), ios::cur);

df.write((char\*)&L, sizeof(L));

}

df.close();

}