Prajapati Yash. P. 20162121023 (BDA)

Date / Page 1

ESFP-2 Assignment-2

(1) Write a function called reversit() that reverses a C-string (an array of char). Use a for loop that swaps the first & last characters, then the second and next-to-last characters, and so on. The string should be passed to reversit() as an argument.

Ans

```
#include <iostream>
#include <cstring>

using namespace std;

void reversit(char str[])
{
    char ch;
    int len, j;
    len = strlen(str);
    j = len-1;
    for(int i=0; i<len/2; ++i)
    {
        ch = str[i];
        str[i] = str[j];
        str[j] = ch;
        j--;
    }
}

int main(){
    char str[20];
```

```
    cout << "\n Enter string : " << endl;
    gets (str);
    reversit (str);
    cout << "\n Reversed string : " << str;
    return 0;
}
```

A(2) (i) Drug object → 40 bytes
Pain reliever → 118 bytes

(ii) Drug :: enterdrugdetails ()
Drug :: void showdrugdetails ()
Tablet :: entertabletdetails ()
Tablet :: showtabletdetails ()
PainReliver :: enterdetails ()
PainReliver :: showdetails ()

(iii) Data Members :-
↳ Tablet :: tablet_name [30];
↳ Tablet :: volume_label [20];
↳ Tablet :: price;

Member Functions :-
↳ Drug :: enterdrugdetails ()
↳ Drug :: showdrug details ()
↳ Tablet :: entertabletdetails ()
↳ Tablet :: showtable details ()

(iv) Data members :- Tablet :: price.

(3) Write a program to arrange subject names in ascending order for class subject with use of pointers of pointers.

Soln

```
#include <iostream>
#include <cstring>
using namespace std;

class subject
{
    char sub[3][10];
    char (*ptr)[10] = sub;
    public:
        void input()
        {
            cout << "Enter 3 subject names:" << endl;
            for(int i=0; i<3 ;++i)
            {
                cout << "subject: " << i+1 << " : ";
                gets(sub[i]);
            }
        }

        void sort()
        {
            char ch[10];
            for(int i=0 ; i<3 ; ++i)
            {
                for(int j=0; j<3 ; ++j)
                {
                    if(strcmp(*(ptr+j),*(ptr+i))>0)
```

```
        {
            strcpy (ch, *(ptr+i));
            strcpy (*(ptr+i), *(ptr+j));
            strcpy (*(ptr+j), ch);
        }
    }
  }
}

void display() {
    for (int i=0; i<3 ; ++i)
        {
            cout << sub[i] << " , ";
        }
    }
};

int main()
{
  subject obj;
  obj.input();
  obj.sort();
  obj.display();

  return 0;
}
```

Prajapati Yash P. 20162121023 (BDA)

Date
Page 5

(4) Write program to create a class of shape .....

Ans

```
#include <iostream>
#include <cmath>

using namespace std;

class shape
{
    protected:
        float aoc, aor, aot, r, l, b, b1, b2, h;

    public:

        virtual void area(){}
        virtual void display(){}
};

class circle : public shape
{
    public:
    void area()
    {
        cout<<"\nEnter radius : ";
        cin >> r;
        aoc = 3.14 * r * r;
    }

void display()
    { cout<<"\nArea of circle : "<< aoc; }};
```

```
class rectangle : public shape
{
  public:
   void area () {
   cout <<"\n Enter length & breadth of rectangle:";
   cin >> l>>b;
   aor = l*b;
  }
void display () {
        cout << "\n Area of rectenge: "<< aor;
        }
    };

class trapezoid : public shape
{
    public:
    void area()
    {
    cout <<"\n Enter height of trapezoid : ";
    cin >>h;
    cout <<" \n Enter length of each base of
          trapezoid:";
    cin >>b1 >>b2;

    aot= ((b1+b2)/2) * h;
   }
void display() {
    cout <<"\n Area of trapezoid: "<< aot;
   }
};
```

```
int main ()
{
    shape *shapeptr;
    shape S;
    shapeptr = &S;
    shapeptr -> area ();
    shapeptr -> display ();
    circle C;
    rectangle R;
    trapezoid T;

    shapeptr = &C;
    shapeptr -> area ();
    shapeptr -> display ();

    shapeptr = &R;
    shapeptr -> area ();
    shapeptr -> display ();

    shapeptr = &T;
    shape ptr -> area ();
    shapeptr -> display ();

    return 0;
}
```

Prajapati Yash.P. 201621210 23 (BDA)

Date
Page 8

(5) Create a Distance class with appropriate member functions.

Ans
```
#include <iostream>
#include <fstream>
using namespace std;

class Distance {
    private:
    int feet;
    float inches;
  public:
Distance() { }
Distance(int ft, float in) : feet(ft), inches(in)

void getdist()
 {
    cout << "\n Enter feet: ";
    cin >> feet;
    cout << "\n Enter inches: ";
    cin >> inches;
 }

void showdist()
    {
      cout << feet << "'\n-" << inches << '\"';
    }
};
```

Prajapati Yash. P. 20162121023 (BDA)

Date
Page 9

```
int main() {
  char ch;
  Distance obj;
  fstream file;

file.open("distance.txt", ios::app | ios::out | ios::in);
do
{
  cout << "\n Distance ";
  obj.getdist();
  file.write((char*)&obj, sizeof(obj));
  cout << "Enter another distance (y/n)? ";
  cin >> ch;
 }
 while (ch=='y');
 file.seekg(0);
 file.read((char*)&obj, sizeof(obj));

 int count=0;
 while (!file.eof())
   {
     cout << "\n Distance " << ++count << ":";
 obj.showdist();
 file.read((char*)&obj, sizeof(obj));
   }

 cout << endl;
 return 0;
}
```

(6) Create a class to add Two times provided in hour minute format.

Sol^n

```
#Include <iostream>
using namespace std;

class Time
{
    int hour1, min1, hour2, min2, hour, min;
    public:

    void input(){
      cout << "Enter first time :- " <<endl;
      cout << "Hours: ";
      cin >> hour1;
      cout << "Minutes: ";
      cin >> min1;
      cout << "Enter second time :- " <<endl;
      cout << "Hours: ";
      cin >> hour2;
      cout << "minutes: ";
      cin >> min2;
    }

    void sum()
    {
      min = min1 + min2;
      hour = hour1 + hour2 + (min / 60);
      min = min % 60;
    }
```

```
void display()
  {
   cout << "Total time is: " << hour << " hours " << mins
                                << " mins";
  }
};

int main()
{
   Time obj;
   obj.input();
   obj.sum();
   obj.display();

   return 0;
}
```