

1.1 Write the output or the error of the following programs. You can write output if there is no error and if there is any error properly mention that error. Writing both error and output will result in deduction of marks [3 marks]

```
i) #include <iostream>
using namespace std;

class Base
{
    protected:
    Base(){
        cout<<"Base constructor"<<endl;
    }
    void show(){
        cout<<"in class A show"<<endl;
    }
    void access(){
        show();
    }
};

class Derived: public Base
{
    public:
    Derived(){
        cout<<"Derived constructor"<<endl;
    }
    void access(){
        Base::access();
    }
};

class NDerived: public Derived
{
    public:
    NDerived(){
        cout<<"NDerived constructor"<<endl;
    }
};

int main()
{
    NDerived obj;
    obj.access();
    return 0;
}
```

Output:

Base constructor  
Derived constructor  
NDerived constructor  
in class A show

03

ii)	<pre>#include &lt;iostream&gt; using namespace std;  class A {     public:     A(int k){         cout&lt;&lt;k&lt;&lt;endl;} }; class B: public A {     public:     B():A(3){         cout&lt;&lt;"B"&lt;&lt;endl;} }; class C: public A {     public:     C():A(5){         cout&lt;&lt;"C"&lt;&lt;endl;} }; class D: public B, public C {     public:     D(){         cout&lt;&lt;"D"&lt;&lt;endl;} };  int main() {     D obj;     return 0; }</pre>	<p>Output:</p> <p>3 ✓ B ✓ 5 ✓ C ✓ D ✓</p> <p>03</p>
iii)	<pre>#include &lt;iostream&gt; using namespace std;  int main() {     char *n;     char v[10];     cin&gt;&gt;v; //suppose input "Usman"     n=v;     cout&lt;&lt;n[2];     return 0; }</pre>	<p>Output:</p> <p>m</p> <p>03</p>

iv)	<pre> #include &lt;iostream&gt; using namespace std;  class A {     A **p;     int var; public:     A(){         var=5;}     A(int var){         this-&gt;var=var;         cout&lt;&lt;this-&gt;var;}     A(A **var, const int k):p(var){         this-&gt;var=k;         cout&lt;&lt;this-&gt;var;}     A&amp; DEF() {         return **p;}     void display(); }; void A::display(){      cout&lt;&lt;var;}  int main() {     A o1, o2(4);     A *pobj=&amp;o1;     A o3(&amp;pobj, 3);     o3.DEF() = o2; // o1 = o2     o3.display();     return 0; } </pre>	<p>Output:</p> <p>4 ✓</p> <p>3 ✓</p> <p>3 ✓</p> <p>28 ✓</p>
-----	--	---

## National University of Computer and Emerging Sciences

FAST School of Computing

Fall-2022

Islamabad Campus

<pre>x) #include &lt;iostream&gt; using namespace std; class A {     public:     ~A(){         cout&lt;&lt;"A"&lt;&lt;endl;} }; class B: public A {     public:     ~B(){         cout&lt;&lt;"B"&lt;&lt;endl;} }; class C: public B {     public:     ~C() {         cout&lt;&lt;"C"&lt;&lt;endl;} }; int main() {     A *ptr=new C;     delete ptr;     return 0; }</pre>	<p>Output:</p> <p><del>A</del> 03</p>
<pre>x0) #include &lt;iostream&gt; using namespace std;  class A {     int a;     public:     A(){         a=5; }     static void display(int a){         cout&lt;&lt;a;} };  int main() {     A obj;     obj.display(15);     return 0; }</pre>	<p>Output:</p> <p>15 / 03</p>

	Output:
<pre> vii) #include &lt;iostream&gt;       using namespace std;        class A       {           int *p;            public:           A();           A(int);           A(A&amp;);           void display();           void setter(int);       };        A::A(){}        A::A(int a){           p=new int;           *p=a;}        A::A(A &amp;temp){           p=new int;           *(this-&gt;p)=*(temp.p);}        void A::display(){           cout&lt;&lt;*(this-&gt;p)&lt;&lt;endl; }        void A::setter(int k){           *(this-&gt;p)=k;}        int main()       {           A obj1(25); *p=25           A obj2=obj1;           obj1.display();           obj2.display();           obj2.setter(2);           obj1.display();           obj2.display();           return 0;       } </pre>	<p>25 25 2 2</p> <p>25</p>

## National University of Computer and Emerging Sciences

FAST School of Computing

Fall-2022

Islamabad Campus

<p>viii) #include &lt;iostream&gt; using namespace std;</p> <pre> class A {     int a;     public:     A(int n):a(n){     }     void display() const{         cout&lt;&lt;a;     }     int getter(){         return a;     }     void setter(int k){         a=k;     } };  class B {     int a;     public:     B(int n):a(n){     }     void display() const{         cout&lt;&lt;a;     }     int getter(){         return a;     } };  class C {     int a;     public:     C(int n):a(n){     }     void display() const{         cout&lt;&lt;a;     }     int getter(){         return a;     } };  A adder(C &amp;obj1, B &amp;obj){     A temp(0);     temp.setter(obj1.getter()+obj.getter());     return temp; } </pre>	<p>Output:</p> <p>03</p> <p>void datatype function cannot be constant as it returns nothing which can be constant.</p>
---	--



ix)	<pre> #include &lt;iostream&gt; using namespace std;  class A {     int a; public:     A(){         a=0;         cout&lt;&lt;"default constructor A"&lt;&lt;endl;}      A(int a){         this-&gt;a=a;         cout&lt;&lt;"para constructor A"&lt;&lt;endl;}      void Adisplay(){         cout&lt;&lt;"A class "&lt;&lt;a&lt;&lt;endl;} };  class B: public A {     int b; public:     B(){         b=0;         cout&lt;&lt;"default constructor B"&lt;&lt;endl;}      B(int b):A(b){         this-&gt;b=b;         cout&lt;&lt;"para constructor B"&lt;&lt;endl;}      void Bdisplay(){         cout&lt;&lt;"B class "&lt;&lt;b&lt;&lt;endl;} };  class C: public B {     int c; public:     C(){         c=0;         cout&lt;&lt;"default constructor C"&lt;&lt;endl;}      C(int c):B(c){         this-&gt;c=c;         cout&lt;&lt;"para constructor C"&lt;&lt;endl;}      void Cdisplay(){         cout&lt;&lt;"C class "&lt;&lt;c&lt;&lt;endl;} }; </pre>	<p><b>Output:</b></p> <p>default constructor A          default constructor B          para constructor A          para constructor B          para constructor A          para constructor B          para constructor C          B class 5</p> <p style="text-align: center;">63</p>
-----	--	--

## National University of Computer and Emerging Sciences

FAST School of Computing

Fall-2022

Islamabad Campus

<pre> int main() {     B obj1, obj2(5);     C obj(5);     obj.Bdisplay();     return 0; } </pre>	
<p>x)</p> <pre> #include &lt;iostream&gt; using namespace std; class Dummy { float z; int x, y; public:     Dummy(int x=0, int y=1):x(x+2),y(y+3){         z = x + y + 1;     }     void print(){         cout&lt;&lt; " X= " &lt;&lt; x         &lt;&lt; " Y = " &lt;&lt; y         &lt;&lt; " Z = " &lt;&lt; z;     } }; int main() {     Dummy d(10);     d.print();     return 0; } </pre>	<p>Output</p> <p>X = 12 Y = 4 Z = 17</p> <p>02</p>

1.2 Write the output or the error of the following programs. You can write output if there is no error and if there is any error properly mention that error. Writing both error and output will result in deduction of marks [5x3 marks]

<pre> #include&lt;iostream&gt; using namespace std;  int* mystery(int* p){     int i;      for (i = 0; i &lt; 2; i++) {         int temp = *(p + i);         *(p + i) = *(p + 4 - i);         *(p + 4 - i) = temp;     }      return p; } </pre>	
--	--

$p[0] = 5$   
 $p[4] = 1$   
 $p[1] = 4$   
 $p[3] = 2$



<pre>int main(){ int x[5] = { 1, 2, 3, 4, 5 }; int i, *p; p = mystery (x); for (i = 0; i &lt; 5; i++) cout &lt;&lt; *(p + i) &lt;&lt; " "; return 0;</pre>	<p>5 4 3 2 1</p> <p>05</p>
<pre>#include &lt;iostream&gt; using namespace std;  int fun(int count){     cout&lt;&lt; count&lt;&lt;endl;     if (count &lt; 4)     {         fun(fun(fun(++count)));     }     return count; }  int main(){     cout &lt;&lt; fun(2);     return 0; }</pre>	<p>2 3 4 4 4 4 4 4</p> <p>4</p> <p>4</p>
<pre>#include &lt;iostream&gt; using namespace std;  void mystery(char* input,int<sup>20</sup> s, char * output, int<sup>0</sup> s1, int i = 0){     output[s1]= input[i];     if (i == s - 1)         return;     if (input[i] == input[i + 1]){         s1++;         output[s1] = '*';     }     mystery(input,s, output,s1+1, i + 1); }  int main(){     char input[20] = "programming letter";     char output[30];     mystery(input,20,output,0);     cout &lt;&lt; output &lt;&lt; endl;     return 0; }</pre>	<p>program*</p> <p>25</p>

## National University of Computer and Emerging Sciences

FAST School of Computing

Fall 2022

Islamabad Campus

## Question 2 [45 Marks]

2.1 Show the output of the following code. In case of an error(s) identify them all. [10 marks]

```

#include <iostream>
using namespace std;
class A {
public:
    A() {}
    virtual double solution() = 0;
private:
    int i;
};
class B : public A {
private:
    int j;
};
class C {
public:
    int f(int a) { return x * a; }
protected:
    void setX(int a) { x = a; }
    int getX() { return x; }
private:
    int x;
};
class D : public C {
private:
    int z;
};
int main() {
    A objA;
    B objB;
    C objC;
    D objD;

    objC.setX(5);
    cout << objC.getX();

    objD.setX(19);
    objD.f(36);
    return 0;
}

```

[ERRORS]

because it has a pure function. It cannot have objects.

B is also an abstract class because it does not override all virtual functions of class A. So, it also cannot have objects.

Output:

There is no error, following will be the output.

50  
19  
-16  
34  
57  
604

## National University of Computer and Emerging Sciences

FAST School of Computing

Fall-2022

Islamabad Campus

```

class H {
public:
    void func() {
        cout << "H's func" << endl;
    }
    void myFunc() {
        int a = 0;
    }
    ~H() {
        cout << "H's destructor" << endl;
    }
};

```

```

class C : public B, public H {
public:
    void func() {
        a *= 3; // 1x3
        A::func();
        cout << "C's func" << endl;
    }
    void myFunc() {
        a += 5;
        b -= 5;
    }
    ~C() {
        cout << "C's destructor" << endl;
    }
};

```

```

void main() {

    A* a = new C;
    a->func();
    cout<<a->getA() << endl;
    delete a;
    a = new B;
    a->func();
    a->myFunc();
    cout << a->getA() << endl;
    B* bptr = new C;
    bptr->func();
    cout<<bptr->getB()<<endl;
    bptr->myFunc();
    cout << bptr->getB() << endl;
    delete a;
    delete bptr;
}

```

A's func  
 C's func  
 H's func  
 B's destructor  
 A's destructor  
 B's func  
 0  
 A's func  
 C's func  
 H's func  
 10  
 B's destructor  
 A's destructor  
 B's destructor  
 A's destructor

a=4  
 a=2  
 b=9  
 a=0

Output:

A's func  
C's func  
H's func  
4  
B's destructor  
A's destructor  
B's func  
0  
A's func  
C's func  
H's func  
10  
B's destructor  
A's destructor  
B's destructor  
A's destructor

12

2.3 There ARE ERROR(S) in the given code. Identify the errors, remove them and show output. [15 marks]

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

class Engine {
protected:
    bool power;
public:
    Engine() {
        cout << "Engine constr" << endl;
    }
    ~Engine() {
        cout << "Engine destr" << endl << endl;
    }
    virtual void turnOn() = 0;
};

class ElectricEngine :public Engine {
public:
    ElectricEngine() {
        cout << "Electric Engine constr" << endl;
    }
    ~ElectricEngine() {
        cout << "Electric Engine destr" << endl << endl;
    }
    void turnOn() { power = 1; }
};

class GasEngine :public Engine {
public:
    GasEngine() {
```

```
Hybrid() {  
    cout << "Hybrid Car constr" << endl;  
}  
~Hybrid() {  
    cout << "Hybrid Car destr" << endl << endl;  
}  
void start(bool mode) {  
    if (mode)  
        Electric::start();  
    else  
        Gasoline::start();  
}  
};  
  
void main() {  
    Hybrid vexel;  
    vexel.start(1);  
    vexel.accelerate();  
    vexel.accelerate();  
    vexel.accelerate();  
    cout << vexel.getSpeed() << endl;  
    vexel.brake();  
    vexel.brake();  
    cout << vexel.getSpeed() << endl;  
}
```

Output:





## National University of Computer and Emerging Sciences

FAST School of Computing

Fall-2022

Islamabad Campus

## Question 3 [45 Marks]

3.1 There is an error in following class definition. Write the name of error and correct the error. [5 marks]

Hint: please note the error is not related to missing library or semicolon etc.

```
class XYZ{
    int x, y;
    const int z;
public:
    XYZ(){
        x=0;
        y=0;
        z=0;
    }
    void display(){
        cout<<x<<y<<z<<endl;
    }
};
int main()
{
    XYZ xyz;
    xyz.display();
    return 0;
}
```

⇒ constant must be initialized in the same line where it is declared.

3.2 Following class definition creates shallow copy of the member variables in both constructors. Modify the class so that it should create deep copy and destroy memory when not needed. [5 marks]

```
class aClass {
private:
    char* a;
    int b;
    int c;

public:
    aClass(char*, int, int);
    aClass(const aClass&);
    ~aClass();
};

aClass::aClass(char* aPtr, int
bVal, int cVal){
    a = aPtr;
    b = bVal;
    c = cVal;
}
delete aPtr;
aPtr = NULL;
```

~~char\* a = new char;~~  
~~char\* aPtr = new char;~~  
 \*a = \*aPtr



```

aClass:: aClass (&aClass ac){
    a = ac.a;
    b = ac.b;
    c = ac.c;
} delete ac.a;
ac.a = NULL;
aClass::~aClass() {
    delete a; a = NULL;
// delete dynamically initialized memory if any
}
    
```

```

a = new char;
ac.a = new char;
*a = *(ac.a);
    
```

3.3 Modify the Dice class with a variable so that each time an object of Dice is created, the variable should increment. Zero marks will be awarded for creating global variables. [5 marks]

```

class Dice{
    int faceValue;
    static int x = 0;

public:
    Dice(int fvalue=0){
        faceValue=fvalue;
        x++;
    }
};
    
```

3.4 Write output of the following program. [5 marks]

```

void flip(int* x, int* y, int*& z)
{
    z = y;
    y = x;
    *x = 200;
}

int main() {
    int i = 10;
    int j = 20;
    int* p = &j;
    flip(&i, &j, p);
    cout << "i is = " << i << endl;
    cout << "j is = " << j << endl;
}
    
```

Output:

i is = 200

j is = 200

p is = 199

# National University of Computer and Emerging Sciences

FAST School of Computing

Fall-2022

Islamabad Campus

```
cout << "p is = " << --(*p) <<
endl;
}
```

3.5 Complete the implementation so that the code given in the main works. [5 marks]

```
class Address {
    int a;
    char arr[25] = "gali 12 H-9 Islamabad";

    a1(int x)
    {
        a = x;
    }
    a1(char b) { cout << func(b); }
    while (arr[i])
    int func(char y)
    {
        for (int i = 0; i < 25; i++)
        {
            if (arr[i] == y)
                return i;
        }
    }
};

int main()
{
    Address a1(15);

    int index = a1('m');

    cout << "Index of 'm' is: " << index;

    return 0;
}
```

3.6 Write a templated function printArrayRange() that should receive an array, count of array elements, lower subscript and higher subscript. It should print array elements in the range. The function should Validate lowSubscript and highSubscript; if either is out of range or if highSubscript is less than or equal to lowSubscript, the overloaded printArray function should display "range error" and terminate. Create a main function and call printArrayRange function for an integer, floating and char type array. [10 marks]

```
template < typename A> typename
A printArrayRange ( A * arr, int n, int low-sub, int high-sub )
{
    for (int i=0; i < n; i++)
    {
        cout << *(arr+i) << endl;
        if (low-sub > high-sub && high-sub > n)
            printArrayRange ( );
    }
    return 0;
}
```

```
A printArrayRange ( )
{ cout << "Range Error"; }
```

```
int main()
{ int count, lower-subscript, higher-subscript;
```

```
cin >> count;
cout << "Enter lower-subscript > higher-subscript";
int arr, count, lower-subscript, higher-subscript;
cout << "Enter count, lower-subscript, higher-subscript";
cin >> count >> lower-subscript >> higher-subscript;
cout << "Enter count, lower-subscript, higher-subscript";
return 0; }
```

3.7 Write output of the following program. [10 marks]

```
void printNow(char *s)
{
    if(*s) {
        cout<<*s;
        printNow(s+1);
    }
}

void printThen(char *s)
{
    if(*s) {
        printThen(s+1);
        cout<<*s;
    }
}

int main()
{
    char *str = "tit for tat";
    printNow(str);
    cout << endl;
    printThen(str);
}
```

Output:

tit for tat

tat ref tit

10

## National University of Computer and Emerging Sciences

FAST School of Computing

Fall 2022

Islamabad Campus

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

## Question 4 [45 Marks]

4.1 Imagine a publishing company that markets both books and audiocassette versions of its works. Create a class Publication that stores the title, and price of a publication. Create two more classes which shows the inheritance with Publication: Book, which adds a page count, and Tape, which adds a playing time in minutes. Each of these three classes should have a getdata() function to get its data from the user at the keyboard, and a putdata() function to display its data. Moreover, ensure these classes supports polymorphism.

You need to write a main function that creates an array of pointers to publication of size 5. In a loop, ask the user for data about a particular book or tape, and use new to create an object of type of Book or Tape to hold the data. Put the pointer to the object in the array. When the user has finished entering the data for all books and tapes, display the resulting data for all books and tapes entered, using a for loop and a single statement to display the data from each object in the array. [10 marks]

```
class Publication {
protected:
    string title;
    int price;
public:
    virtual void getdata();
    virtual void putdata();
    // function definitions?
};
```

```
class Book : public Publication {
    int pagecount;
public:
    void getdata()
    {
        cin >> this->title;
        cin >> this->price;
        cin >> this->pagecount;
    }
    void putdata()
    {
        cout << "Title" << title << endl;
        cout << "Price" << price << endl;
        cout << "pages" << pagecount << endl;
    }
};
```

```
class Tape : public Publication {
    int playingTime;
public:
    void getdata()
    {
        cin >> this->title;
        cin >> this->price;
        cin >> this->pagecount;
    }
    void putdata()
    {
        cout << "Title" << title << endl;
        cout << "Price" << price << endl;
        cout << "playingTime" << pagecount << endl;
    }
};

int main()
{
    Publication ** arr = new Publication[5];
```

7.5



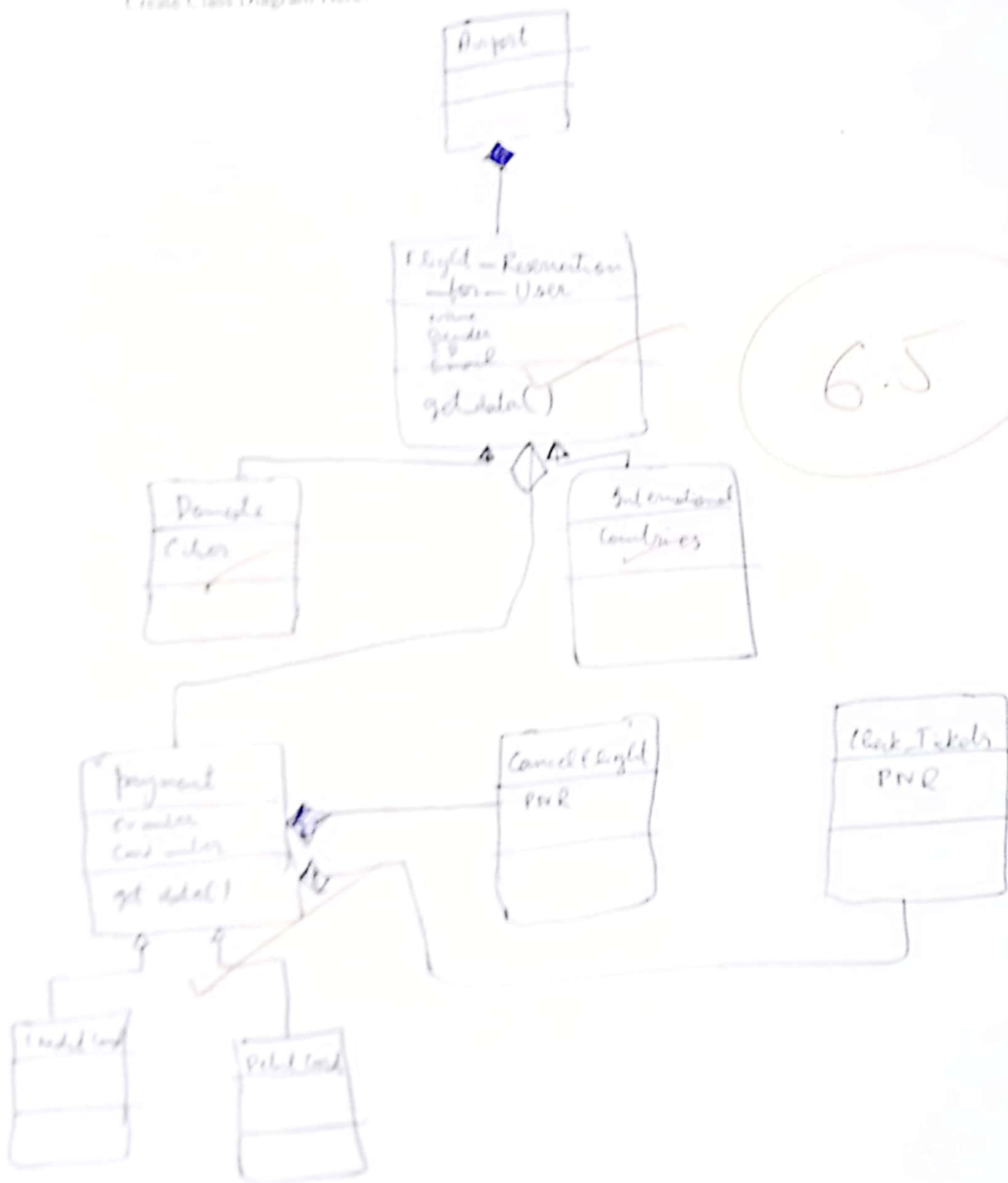
# National University of Computer and Emerging Sciences

FACS School of Computing

Fall-2022

Islamabad Campus

Create Class Diagram Here.



```
class BankManagementSystem {
```

```
    void deposit() = 0;
```

```
};
```

```
class SavingsAccount : public BankManagementSystem {
```

```
    void deposit() over;
```

```
};
```

