Name: Noot-W-Ain Sagheet

Reg#: B(\$ 223020

Date: 10th Nov. 2023

Course: "O O P"

Section: 1

Assignment No:

Description: (Copy Constructors, file handling)

Submitted to: Sis Adnan Jelani"

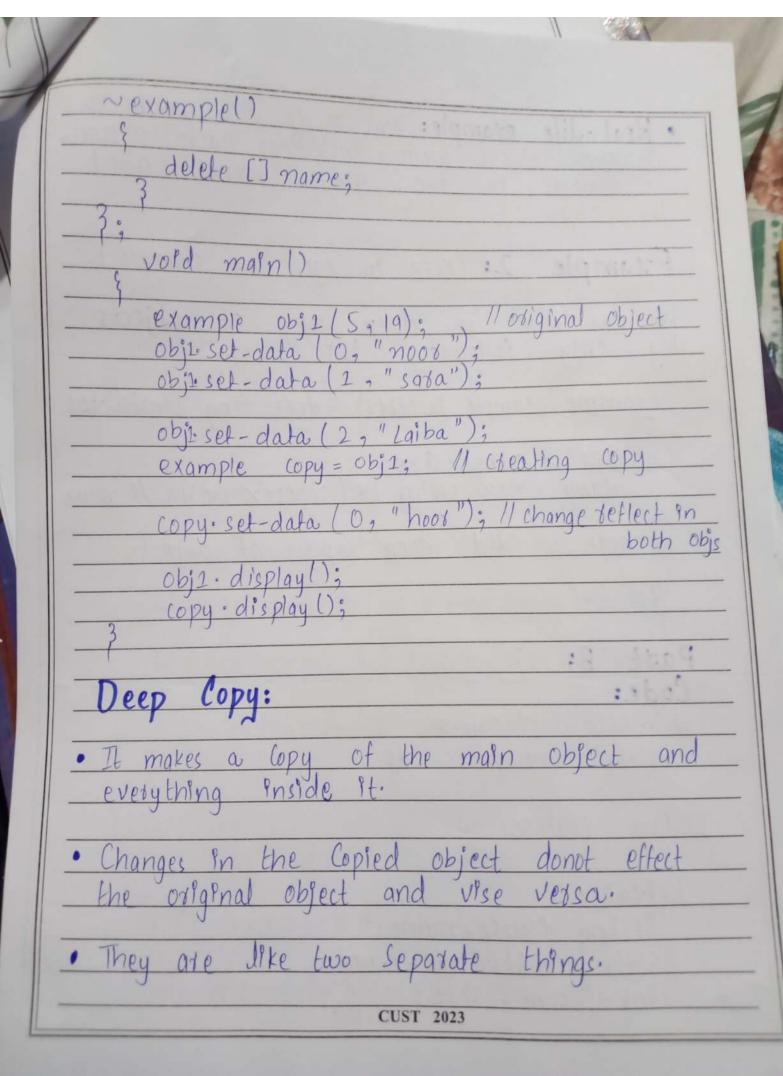


Capital University of Science & Technology

Question # 01:  (Past A)
Question # 01:
(Past A)
Shallow Copy: (In case of arrays)
• It makes a copy of the main object but not the things inside it.
• So, changes in the copied object affect the Original object and vice versa.
Real-life example: Imagine it's like taking a  Photo of a group, and if Someone in the  group changes their shirt, it's reflected in  the photo too.
Example: 1
# include ciostream> using names pace std;
class example
Private:
Int age;
String * name;
Stoing * name;  Int size;
Public:
example (int sq int a)
CUST 2023

•

```
Size = Si
 name = new String [size];
age = 0;
example (example & other) // Shalow Copy Constructor
                      11 9s under use.
    size = other size;
    name = other name;
     age = othet age;
 void set-data (Pnt i, string n)
   if (i>=0 && iLsize)
        name [i] = n;
 vold display!
   For lint i= 0; 1 L size; i++)
      cout « Names: " Ll name [i] Ll";
       continends;
   conter "Age: "Lage;
```

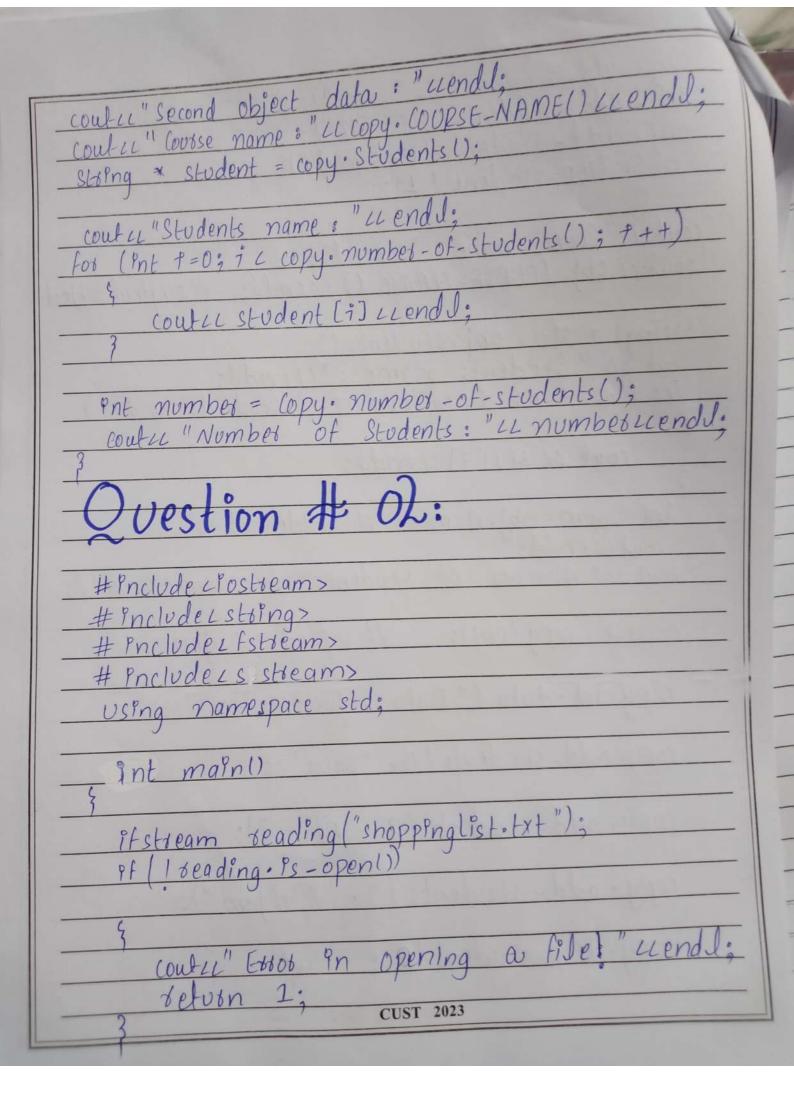


Real-life example: Think of It like making a duplicate of a group, Including each person, affect so changes in one group donot affect the other.
Real-Life example: mink produding each person,
duplicate of a group 2 mous donot affect
so changes en one great
the othet.
Example 2: (keep in eye first example)
We can make a deep copy of objects by only following this step;
example (example & other) // deep copy Constructor
name = new string (* (other name); // now
11 here we did deep copy of objects
3
Past B:
Code:
#include ciostream>
using namespace std;
class course
*
Polvate:
String course-name;  String * students-name;
int Count, Size; Cust 2023

```
Public:
 course (int s, int c)
     count = c; size = s;
     Students - name = new storng[size];
  course (course & other) 1/deep Copy Constructor
   course-name = other. course-name;
   Students-name = new string (* (other. students-name);
   count = other : count;
   stre = other stre;
   void set-data (string c-n
      course-name = c-n;
   void add-students (int index, string name
     9f ( Index >= 0 && Index LSize)
        students - name [index] = name;
        Count++;
   void doop-student (int index)
        (Index >= 0 && Index L count)
         Fot (Int = Index; 7 4 count -1; 1++
```

```
Students-name [i] = Students-name [i+1];
   count -- ;
Storng COURSE - NAME()
  selvan course-name;
 Storng x students ()
    setusn students-name;
  int number- of-students()
   setuon count;
~ course()
 defete [] students-name;
void main ()
covose obj (5,0);
obj. set-data ("OBP");
obj. add-students (0, "noot");
obj. add-students (1, "Laiba");
cust 2023
```

```
obj. add-students (2, "sara");
obj. add-students (3, "Ali");
obj. add-students (4, "Asad");
  Obj. drop-student (1):
cout " Course name: " mendd; cout u Obj. COURSE-NAME () mendd; loriginal object
 Storng * Stu = Obj. Students ();
cout LL "Students name: "Llendd;
for (int i=0; i Lobj. number-of-students(); i++)
       cout LL stu[i] LLendl:
    int num = obj. numbes - of -students ();
conticendd;
conticendd;
    course copy (obj); // Copied object
     copy-set-data (" Data Structures");
     copy. add-students (0, "saba");
     copy. add-students (1, "Amna");
      copy · add - students (2, "Masyam");
      copy.dop-student (0):
                                  CUST 2023
```



	double total-porce = 0.0;
_	contre "Product Details : "mendl;
_	while (get line (teading, line))
_	Storngstream ss(line);
-	double product-price;
	getline (ss, product-name, ','); ss >> product-price;
	conter "Product = "Le product-name, Le" , proce="
	total-price = total-price + product-price;
	3
	contre "Total posce=" LL total-posce. LLends;  seading. close();
1	setvan 0;
	Github Account details:  Github email: Chazeem 19800 @gmail.com  Github I D: NOOR-UL-AIN-SAGHEER.  CUST 2023