İ.T.Ü. 18.06.2001
Computer Engineering Department
Feza BUZLUCA, PhD
Çiğdem ŞENGÜR, BSc

**Object Oriented Programming Final Examination**

**Question 1:**

```
#include <iostream>
#include <string>
using namespace std;
void main()
{
   Array<Person> ArrP(3);     // Person Array with three empty places
   Person p1("Murat", 42);    // Create persons (name, age)
   Person p2("Neşe", 27);
   try{                       // Store into array
      ArrP[2]=p1;
      ArrP[0]=p2;
   }
   catch(const string & msg){  // indices may be out of bounds
     cout << msg << endl;
   }
   string name;
   int age;
   cout << "Give the name of the person to be searched: ";
   cin >> name;
   cout << "Give the age of the person to be searched: ";
   cin >> age;
   Person searched(name,age);        // Create a person for searching
   if(ArrP.find(searched))           // Search in the array
       cout << "found\n";
    else cout << "NOT found\n";
    Person sp = ArrP.smallest();
    sp();                            // print the name and age
    cout << " is the youngest person in the array\n";

    Array<int> ArrI(2);              // Integer Array with two empty places
    try{
      ArrI[0]=7;
      ArrI[1]=-6;
   }
   catch(const string & msg){         // indices may be out of bounds
      cout << msg << endl;
   }

   if(ArrI.find(4))                   // Search for 4
       cout << "found\n";
    else cout << "NOT found\n";
    int k = ArrI.smallest();
    cout << k << " is the smallest number in the array\n";
}
```

**Array** is a template class that can contain different types of elements. It has a pointer to its elements (and other necessary elements).

**Person** is a user-defined class. It contains name (string) and age of a person.

**a)** Design and write the template class **Array** so that it can be used in the given main program.

**b)** Design and write the class **Person** so that it can be used with template class **Array**.

Do not use containers and algorithms of the STL.

**Question2:**
A program to model the kinds of people one finds in a university will be written. The categories are **`students`**, **`teachers`** and **`assistants`**. All these categories are *kinds of* **`person`**. An assistant is a student and a teacher as well.
A `person` has a name (`string`). A `student` has grade point average (GPA), the `teacher` has the number of scholarly papers he/she has published and the `assistant` has the number of courses he/she assists in addition to other data members.
All the classes contain constructors to fill all data spaces. The **`print()`** function prints all data members of an object on the screen. They also contain a function called **`isSuccessful()`**. Students with a GPA over 3.5,
teachers who have published more than 50 papers and
assistants who assist more than 3 courses and have a GPA over 3.5 are considered successful.
This function returns **`true`** if someone is successful or **`false`** if not.

In `main()`, define an array of pointers, which can point to students, teachers and assistants. The program first asks the kind of person and then lets the user enter the name. For students, the program also asks for the GPA; for teachers, it asks for the number of publications and for assistants it asks the number of courses additionally. The maximum number of pointers to objects, which can be kept in the array, is 100. When the user is finished, the program prints out the names and other data for all persons. It also prints whether a person is successful or not. Here's some sample interaction:

```
Enter student,teacher,assistant (s/t/a): s
   Enter name: Timmy
   Enter student's GPA: 1.2
   Enter another (y/n)? y
Enter student,teacher,assistant (s/t/a): t
   Enter name: Shipley
   Enter number of teacher's publications: 75
   Enter another (y/n)? y
Enter student,teacher,assistant (s/t/a): a
   Enter name: Brenda
   Enter student's GPA: 3.9
   Enter number of teacher's publications: 15
   Enter number of assisted courses: 4
   Enter another (y/n)? n


Name = Timmy
   GPA = 1.2
Name = Shipley
   Publications = 75
   (This person is successful)
Name = Brenda
   GPA = 3.9
   Publications = 15
   Num. of courses = 4
   (This person is successful)
```

**a)** Write all necessary classes (including their methods).
**b)** Write and explain the main program.