Return your answers in Tuubi before deadline!

1. **Function Template to find the largest number**. Next you have to implement program to display largest among two numbers using function templates. Implement template function **Large** and use it in main function where you define variables

```
int i1, i2;
float f1, f2;
char c1, c2;
```

After that you have to make statements so that the output is like in figure 1.

```
Enter two integers:
5
10
10 is larger.

Enter two floating-point numbers:
12.4
10.2
12.4 is larger.

Enter two characters:
z
Z
z has larger ASCII value.
```

Figure 1. Sample print in Dev C++ -program

< contined >

2. **Simple calculator using class template**. Next you have to implement program to add, subtract, multiply and divide two numbers using class template. Implement template function **Calculator** and it in main function where you make calls

```
Calculator<int> intCalc(2, 1);
Calculator<float> floatCalc(2.4, 1.2);
```

After that you have to make statements so that the output is like in figure 2.

```
Int results:
Numbers are: 2 and 1.
Addition is: 3
Subtraction is: 1
Product is: 2
Division is: 2

Float results:
Numbers are: 2.4 and 1.2.
Addition is: 3.6
Subtraction is: 1.2
Product is: 2.88
Division is: 2
```

Figure 2. Sample print in Dev C++ -program

< contined >

3. **Virtual Function**. Implement a **person** class which contains protected property **name**. This class contains functions **getName** and **putName**. Function **getName** requests **name** and function **putName** prints the name. This class also contains two pure virtual functions virtual void **getData**() and virtual bool **isOutstanding**. Implement derived class **student** which contains protected property **gpa** (grade point average). Implement function **getData** which get student data from user. This function inherits **getName** of base class. Implement also function **isOutstanding**, which returns true if gpa is greater than 3.5. Implement derived class **professor** which contains protected property **numPubs** (number of papers published). Implement function **getData** which get professor data from user. This function inherits **getName** of base class. Implement also function **isOutstanding**, which returns true if gpa is greater than 100. In main function define an array of pointers to persons. Make a loop where you can input data of professor and/or students like in figure 3. In the end program prints content of array like in figure 3.

```
Enter student or professor (s/p): s
   Enter name: Timmy
   Enter student's GPA: 1.2
   Enter another (y/n)? y
Enter student or professor (s/p): s
   Enter name: Brenda
   Enter student's GPA: 3.9
   Enter another (y/n)? y
Enter student or professor (s/p): s
   Enter name: Sandy
   Enter student's GPA: 2.4
   Enter another (y/n)? y
Enter student or professor (s/p): p
   Enter name: Shipley
   Enter number of professor's publications: 714
   Enter another (y/n)? y
Enter student or professor (s/p): p
   Enter name: Wainright
   Enter number of professor's publications: 13
   Enter another (y/n)? n

Name is: Timmy
Name is: Brenda
   This person is outstanding
Name is: Sandy
Name is: Shipley
   This person is outstanding
Name is: Wainright
```

Figure 3. Sample print in Dev C++ -program [Laf02, p. 514]

< continued >

4. **Function Templates with Multiple Arguments**. You have to implement a function template. The purpose of this function is to search an array for a specific value. The function returns the array index for that value if it finds it, or –1 if it can't find it. The arguments are a pointer to the array, the value to search for, and the size of the array. In main() we define four different arrays of different types, and four values to search for. You have to treat type char as a number. Then we call the template function once for each array. Sample output is in figure 4.

```
 5 in chrArray: index=2
 6 in intArray: index=-1
11 in lonArray: index=4
 4 in dubArray: index=-1
```

Figure 4. Sample print in Dev C++ -program