Instructions: This exam is to be taken in silence, without notes, books, or electronic devices (including "smart" watches or earbuds). The time limit to complete it is the duration of the class period (1 hour and 15 minutes). Answer the following questions and problems, trying to be as clear and as accurate as possible. Take your time to read the statements carefully before trying to answer them. If you need more space, write on the back of your test page and indicate it clearly. When writing code, make sure your special punctuation characters are legible, and your lowercase and uppercase letters are easy to distinguish. As usual, every statement or series of statement is assumed to be in a valid class and method, and you can use the `C.W( )` and `C.WL( )` abbreviations.

____ / 10 pts. **Problem 1** Consider the following code:

```
string DisplayH(int[] aP, int currentIndex)
{
  string ret = "";
  if (aP.Length == currentIndex + 1)
    ret += " || " + aP[currentIndex].ToString();
  else ret+= DisplayH(aP, currentIndex+1) + ", " +
   ↪  aP[currentIndex].ToString();
  if (currentIndex == 0){ret += " ** "; }
  return ret;
}
string Display(int[] aP)
{
    return DisplayH(aP, 0);
}
```

Which `string` would be returned if `arrayS` contains values `1`, `2` and `4` and `Display(arrayS)` is called? You can justify your reasoning by walking through the function calls.

____ / **30 pts.** **Problem 2** Suppose that at `fPath` is located a file containing text. Write a program that

- Reads the content of the file, and sums all the numerical values when they are the only content on the line.
- Writes at the end of the file "Total:" followed by the total obtained previously.

For example, the following file on the left would be edited as shown on the right once the program is done.

```
File at fPath (before)
┌──────────────────┐
│12.01             │
│15.52             │
│12.48             │
└──────────────────┘
```

```
File at fPath (after)
┌──────────────────┐
│12.01             │
│15.52             │
│12.48             │
│Total: 40.01      │
└──────────────────┘
```

In addition, your program should

1. handle possible issues (file missing, not enough memory, …) graciously,
2. handle lines containing text: a line containing "This is 1 test" should simply be ignored.

____ / 30 pts. **Problem 3** Consider the UML class diagram displayed on the last page (p. 6).

1. Write the complete implementation of the `Shape` abstract class. The `ToString` method should simply return the string `"This shape is "`.

2. Write an implementation for the `Radius` property of the `Circle` class such that setting the radius to a negative value would result in an `ArgumentOutOfRangeException` (that you can shorten to `AOORE`) exception being thrown. Add an attribute if needed.

3. Write the `Diameter` property for the `Circle` class, which should return 2 times the radius. Only the `get` should be provided: briefly explain why the `set` is missing.

4. Write an implementation for the `ToString` method of the `Rectangle` class that returns a `string` containing what was returned by the `Shape`'s `ToString` method, the width, length and area of the calling object. For example, for a `Rectangle` with width 10 and length 5, it should be of the form "This shape is a rectangle (W: 10, L: 5, Area: 50)".

5. Write the `Equals` method for the `Rectangle` class. It should return **`true`** if the calling object and the parameter have the same lengths and same widths, or if one can be obtained by rotating the other.

6. Briefly explain how `Shape` could be converted into an interface and what would be the benefit(s) and disadvantage(s) of carrying out such a modification.

____ / 15 pts. **Problem 4** Answer the following short questions (check all that applies):

1. A method …

   ☐ … can be accessed from any class if it is marked **`protected`**.
   ☐ … can be accessed from its derived classes if it is marked **`private`**.
   ☐ … is recursive if it terminates.
   ☐ … is recursive if it calls itself.
   ☐ … must be overridden if it is marked **`abstract`**.
   ☐ … cannot be overriden if it is marked **`sealed`**.

2. A method with header **`public virtual`** `void Test(int a, ref int b)` will…

   ☐ Return a value
   ☐ Set the value of `a`
   ☐ Set the value of `b`
   ☐ Assume that `a` has a value
   ☐ Assume that `b` has a value
   ☐ Be impossible to override

3. A method is forced to set the value of an `int` argument `c` passed as a reference, if it has…

   ☐ … **`out`** `int c` in its header
   ☐ … `int` **`out`** `c` in its header
   ☐ … **`ref`** `int c` in its header
   ☐ … `int` **`ref`** `c` in its header

____ / 5 pts. **Problem 5** Name two methods in the `String` class, and give 1. their return type, 2. their argument datatype(s), 3. a brief explanation of what they do.

Consider the following diagram to answer Problem 3: