

Examination #5

Name: _____ **Instructor's Solution** _____ SUID: _____

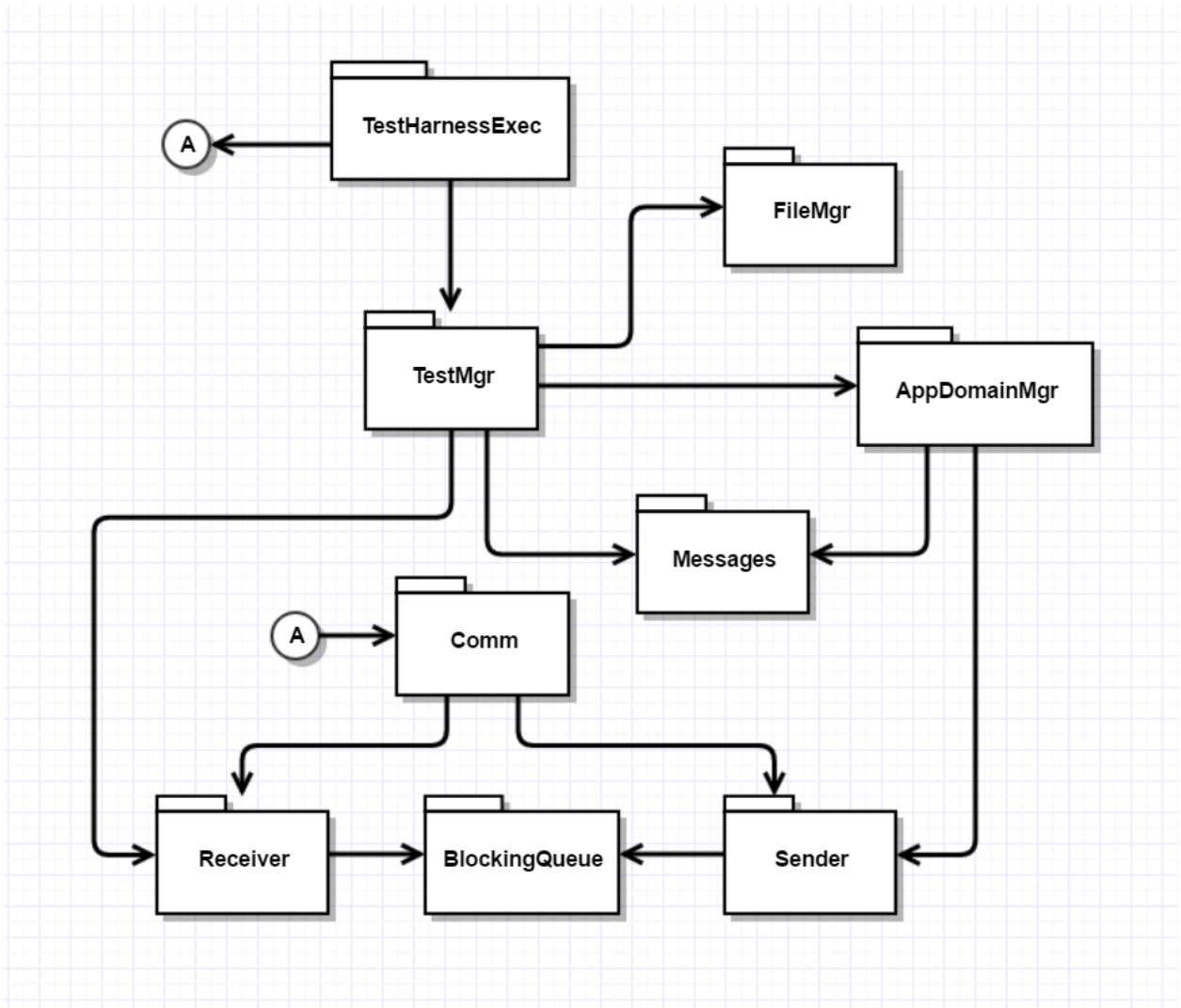
This is a closed book examination. Please place all your books on the floor beside you. You may keep one page of notes on your desktop in addition to this exam package. All examinations will be collected promptly at the end of the class period. Please be prepared to quickly hand in your examination at that time.

If you have any questions, please do not leave your seat. Raise your hand and I will come to your desk to discuss your question. I will answer all questions about the meaning of the wording of any question. I may choose not to answer other questions.

You will find it helpful to review all questions before beginning. All questions are given equal weight for grading, but not all questions have the same difficulty. Therefore, it is very much to your advantage to answer first those questions you believe to be easiest.

1. Draw a package diagram for the TestHarness of Project #4.

Answer:



2. Suppose that the TestHarness, of Project #4, uses a File Service with a service method:

```
Stream downloadFile(string filename);
```

Write all the code for creating, starting, and running a thread that uses this method to acquire the files it needs from the Repository, to run a test, assuming that the filenames are stored in a List<string>. Note: you don't need to write any code for the Repository.

Answer:

```
void download(string filename)
{
    int totalBytes = 0;
    try
    {
        Stream strm = channel.downloadFile(filename);
        string rfilename = Path.Combine(SavePath, filename);
        if (!Directory.Exists(SavePath))
            Directory.CreateDirectory(SavePath);
        FileStream outputStream = new FileStream(rfilename, FileMode.Create);

        Action dwnld = () =>
        {
            hrt.Start();
            while (true)
            {
                int bytesRead = strm.Read(block, 0, BlockSize);
                totalBytes += bytesRead;
                if (bytesRead > 0)
                    outputStream.Write(block, 0, bytesRead);
                else
                    break;
            }
            hrt.Stop();
            ulong time = hrt.ElapsedMicroseconds;
            Console.WriteLine("\n Received file \"{0}\" of {1} bytes in {2} microsec.",
                filename, totalBytes, time
            );
        };
        ThreadStart ts = new ThreadStart(dwnld);
        Thread t = new Thread(ts);
        t.Start();
        threads.Add(t);
    }
    catch (Exception ex)
    {
        Console.WriteLine("\n {0}", ex.Message);
    }
}
```

3. What is a Repository, and why is its use an important part of the Software Collaboration Federation you will discuss in Project #5?

Answer:

A Repository, as defined for SMA projects this semester, is a storage and control mechanism for a Project's code baseline. Its importance is defined by the services it provides:

- Protected storage of a project's code
- Efficient access to a component and all its dependencies simply by naming the component
- Browsing content:
 - Metadata descriptions
 - Dependency relationships
 - Source code contents
- Support for plugins that provide versioning policies, ownership policies, check-in and check-out policies, and code quality analysis

4. Write a code fragment to define a lambda and bind to an Action delegate, where the lambda calls a function `string convert(int i) { ... }`. You may assume the `convert` function is a public member of the class in which the lambda is being defined. You are not required to provide the code for `convert`, but just use it to define the lambda's processing. Furthermore, the Action delegate has no generic parameters.

Answer:

```
class MT5Q4
{
    public static string convert(int i)
    {
        return Math.Abs(i).ToString();
    }
    static void Main(string[] args)
    {
        Console.WriteLine("\n MT5Q4 - lambda capture");
        Console.WriteLine("\n =====\n");

        for (int i = -2; i < 3; ++i)
        {
            string result = "";
            Action act = () =>
            {
                result = MT5Q4.convert(i);
            };
            act.Invoke();
            Console.WriteLine("\n {0} converts to \"{1}\"", i, result);
        }
        Console.WriteLine("\n\n");
    }
}
```

5. What does the term "Software Architecture" mean? What are the most important things to discuss when talking about the architecture of some software system?

Answer:

Software Architecture is the structure, concepts, and organizing principles that define what a software system does, how clients interact with it, the results it produces, and benefits and risks that may be associated with specific design choices. It is not concerned with detailed requirements or implementations.

Its purpose is to simplify a complex system, by eliminating implementation details, and providing diagrams and explanations so it can be easily understood. By defining an architecture early in a project, its architects can reason critically about its details before ever committing to large implementation efforts.

An architectural document will normally address:

- Goals – main objectives of the system
- Uses – how people and other software will interact with the system
- Tasks – activities of system and its major partitions
- Partitions – packages that make up the system and their responsibilities
- Interactions – relationships and data flows between partitions
- Events – any occurrence that affects system activities
- Views – appearance of the system to users and its designers
- Performance – efficient use of system resources, e.g., CPU, Network, Memory

6. WCF clients use proxies for remote service objects to enable the client to use the remote object. Assuming that Message instances have a "To" property to define the destination url, write code for a send thread that dequeues messages, examines their "To" property, and if different than any previously used url, creates a new proxy, using a method IComm CreateProxy(string url), saves the url and proxy in an appropriate .Net container, and uses the new proxy to send the message. Before attempting to send any message it checks the container to see if a proxy for that url has already been created, and if so, uses the existing proxy.

Answer:

```
Dictionary<string, ICommunicator> proxyMap =
    new Dictionary<string, ICommunicator>();

void ThreadProc()
{
    tryCount = 0;
    while (true)
    {
        Message msg = sndBlockingQ.deQ();
        if(!proxyMap.ContainsKey(msg.to))
        {
            proxyMap[msg.to] = CreateSendChannel(msg.to);
        }
        while (true)
        {
            try
            {
                proxyMap[msg.to].PostMessage(msg);
                Console.WriteLine("\n posted message from {0} to {1}", name, msg.to);
                tryCount = 0;
                break;
            }
            catch (Exception ex)
            {
                Console.WriteLine("\n connection failed");
                if (++tryCount < MaxCount)
                    Thread.Sleep(100);
                else
                {
                    Console.WriteLine("\n {0}", "can't connect\n");
                    proxyMap.Remove(msg.to);
                    tryCount = 0;
                    break;
                }
            }
        }
    }
    if (msg.body == "quit")
    {
        Console.WriteLine("\n send thread shutting down");
        break;
    }
}
```

7. Explain why we use reflection in the Project #4 TestHarness. You do not need to write any code to answer this question, but may do so if that makes it easier for you to provide your answer.

Answer:

The test harness needs to use an instance of a test driver class from a loaded library. But it has no way of knowing the name of the class or even if a loaded library contains a test driver.

So, it must use reflection to find a class that implements the ITest interface, and create an instance of that class by using reflection, then invoke its test method.