C/C++ Candidate Assessment Solution by Don Herod

The approach utilized to fulfill the requirements involved the creation of a dynamic link library. It was named frstdlib.dll to simulate the creation of a standard library for company FRSS. I took this approach because both circle and time seemed like prime candidates for incorporation into a library containing primitive functionality that could be reused.

I created 3 solutions. FR.StdLib is the solution for the standard library dll. FR.Demo.Circle is the solution for the demonstration of the circle struct. FR.Demo.Time is the solution for the demonstration of the time class. Each of the demonstration solutions contains a project for their respective demonstration as well as the FR.StdLib project. Note that the intent was for flexibility in how the standard library could be utilized for software development with solutions being able to add the FR.StdLib project or projects being able to add a reference to FR.StdLib and link with the dll appropriately.

The 3 rar files delivered should be extracted into directory with all of the solutions at the same level. This will replicate the directory structure that I setup as being ready to integrate with source code control – any of the 3 solutions will be able to access and persist the files in the FR.StdLib project.

\FR.StdLib\FR.StdLib

\FR.Demo.Circle\FR.Demo.Circle

\FR.Demo.Time\FR.DemoTime

The main documentation of the circle struct and time class are in their respective files (FrCircle and FrTime) in the FR.StdLib solution.

I have included them here for detail:

//-------------------------------------------------------------------------------------------------

// FrCircle.cpp

//

// Appropriate copyright here

//

// FrCircle defines a struct in the FrStdLib library that represents a circle.

//

// There are 2 main usages, based upon construction:

//

// 1. Data Structure

// a. FrCircle struct is used as a data structure to hold the qualities of a circle.

// b. It is constructed with the no argument constructor.

// c. It initializes to the center set to (0,0) and the background set to transparent.

// d. Qualities other than center and background are undefined until set.

// e. All qualities are public and can be set to a provided value.

//

// 2. Advanced Initialization

// a. FrCircle struct is used as a completely defined circle.

// b. Is is constructed with the minimum initialization needed, namely the center and radius.

// c. It initializes and calculates all other qualities, with background set to transparent.

// d. No qualitites are initially undefined.

// e. All qualities are public and can be set to a provided value.

//-------------------------------------------------------------------------------------------------

//-------------------------------------------------------------------------------------------------

// FrTime.cpp

//

// Appropriate copyright here

//

// FrTime defines a class in the FrStdLib library that represents a time.

//

// There are 2 types supported:

//

// 1. Time

// a. FrTime objects can be constructed with the no argument constructor and are initialized

// with the current time.

// b. FrTime objects can be constructed with hour, minute, and second, and are

// initialized to that time provided.

// c. There are accessors and mutators provided for hour, minute, and second, and they update

// the FrTime object with the values provided and get the values represented.

// d. The FrTime object can be reset to the current time.

// e. The FrTime object can output a string representing its time.

//

// 2. DateTime

// a. FrTime objects can be constructed with the no argument constructor and are initialized

// with the current time.

// b. FrTime objects can be constructed with hour, minute, second, day, month, and year,

// and are initialized to that datetime provided.

// c. There are accessors and mutators provided for hour, minute, second, day, month, and year,

// and they update the FrTime object with the values provided and get the values represented.

// d. The FrTime object can be reset to the current time.

// e. The FrTime object can output a string representing its datetime.

//-------------------------------------------------------------------------------------------------