How to Freeze and Unfreeze an Animation and Maintain Time Continuity

At the Top of the Program:

```c
const int MS_PER_CYCLE = 10000; // 10000 milliseconds = 10 seconds
float TimeFrozen; // when animation was frozen
float TimeUnfrozen; // when animation was unfrozen
float TimeElapsed; // how much time elapsed between freezing and unfreezing
```

In Reset( ):

```c
TimeElapsed = 0.f;
```

In Keyboard( )

```c
case 'f'
    case 'F':
        Freeze = ! Freeze;
        if ( Freeze )
            glutIdleFunc(NULL);
            TimeFrozen = Time - TimeElapsed;
            // the value of ms is between 0 and MS_PER_CYCLE-1
            Time = (float)ms / (float)MS_PER_CYCLE; // makes the value of Time [0.,1.)
            TimeFrozen = TimeFrozen + 1.f; // wrap-around
        else
            glutIdleFunc(Animate);
            TimeUnfrozen = Time;
            TimeElapsed = TimeUnfrozen - TimeFrozen;
            // the value of ms is between 0 and MS_PER_CYCLE-1
            Time = (float)ms / (float)MS_PER_CYCLE; // makes the value of Time [0.,1.)
            TimeUnfrozen = TimeUnfrozen + 1.f; // wrap-around
        break;
```

When Drawing

```c
float time = Time - TimeElapsed;
if ( time < 0. )
    time = time + 1.f; // wrap-around
```

When drawing, now use `time` in the same way you used `Time` before. For example:

```c
glRotatef( 360.0f * time, 0., 1., 0. );
```

or

```c
float y = Amplitude * sinf( 2.0f * PI * time );
```