PHY 387N: Relativity Theory II Spring 1998 Project 1
Due: Thursday, February 12, 12:30pm

Note: *Both of the following problems require the numerical solutions of sets of ODEs. I recommend that you do all of your "low-level" coding in* `Fortran 77`*, and that you use the* `ODEPACK` *routine* `LSOSA` *to integrate the ODEs. (You may wish to do "higher level" coding using shell programming features, or* `Perl`*; however,* `Fortran` *will also suffice for that purpose.) See the on-line course notes from PHY381C (available via the 387N course home page) for sample programs which use* `LSODA`*, as well as information on* `Unix` *and* `Fortran` *77 programming. In particular, note that the "Course-Related Software" page, also linked to the 387N home page, describes much of the available software which you may find useful (including the plotting packages* `sm` *(*`Supermongo`*—highly recommended) and* `gnuplot`*). If your group gets hung up on any aspect of the assignment, contact the instructor (via e-mail or an office visit) immediately. Finally, all of the programs, data etc. which you "hand in" should reside in your group account. Be sure that you adequately document where various files etc. can be found within that account. See the solution keys to the 381C homework assignments to get an idea of what your submissions should look like—they don't necessarily have to be in two-column format, though!*

Problem 1: Write a program to integrate the equations for geodesic motion of a massive test-particle in the equatorial plane of the Schwarzschild space-time:

$$ds^2 = -\left(1 - \frac{2M}{r}\right) dt^2 + \left(1 - \frac{2M}{r}\right)^{-1} dr^2 + r^2\, d\Omega^2$$

where, as usual, $d\Omega^2$ is the metric on the unit 2-sphere. Your program *must* have the following usage

```
usage: orbit <L2> <R0> <Rdot0> <dtau> <tau_f>
```

where:

- `<L2>` is the square of the particle's angular momentum parameter (see Wald (6.3.13))
- `<R0>` is the initial radius of the particle
- `<Rdot0>` is the initial time derivative (w.r.t. *proper time*) of the particle radius
- `<dtau>` is the proper time interval between each output
- `<tau_f>` is the final proper time of the integration

Note that you are to use the particle's *proper time* as the independent variable in the solution of the geodesic ODEs.

`orbit` *must* produce output (on standard output) suitable for input into the program `pp2d` which you can use to help visualize your integrations. (Type `pp2d -h` on any on the Center SGI machines (except `infeld` and `hoffmann`) for a description of the appropriate format and brief instructions for using `pp2d`. Also note that `pp2d` uses X-based windowing calls, so that you can run it on one workstation (such as `einstein`), and have the visualization window appear on another (such as `infeld`)). The output should describe the fixed black hole, as well as the moving particle (i.e. two particles total), and notice that `pp2d` expects *three* spatial coordinates per particle (i.e. x, y and z) although only two (x and y) are used in the display.

Use your program to generate some "interesting" orbits, including (but not limited to) stable and unstable circular orbits. Thoroughly document the equations of motion you solve, your program and all results.

Problem 2: Consider the spacetime interior to a static, spherical fluid star (as discussed in Wald, 6.2, and in 387M last term):

$$ds^2 = -e^{2\phi(r)}\, dt^2 + \left(1 - \frac{2m(r)}{r}\right)^{-1} dr^2 + r^2\, d\Omega^2$$

A complete set of equations describing hydrostatic equilibrium is

$$\begin{aligned}
\frac{dm}{dr} &= 4\pi r^2 \rho \\
\frac{d\phi}{dr} &= \frac{m + 4\pi r^3 P}{r\,(r-2m)} \\
\frac{dP}{dr} &= -\left(P+\rho\right)\frac{d\phi}{dr} = -\left(P+\rho\right)\frac{m+4\pi r^3 P}{r\,(r-2m)} \\
P &= P\left(\rho\right)
\end{aligned}$$

where $\rho \equiv \rho(r)$ and $P \equiv P(r)$ are the total energy density and pressure of the fluid, respectively (as measured in the fluid rest-frame). For a specific equation of state ($P = P(\rho)$), the above equations can be integrated to generate a *family* of stellar models, parametrized, for example, by the central density $\rho(0)$.

Write a program to solve the above equations for the simple (and unrealistic!) two-parameter family of equation-of-states:

$$P = C\,\rho^{\lambda}$$

where C and λ are constants. The program must have the usage

```
usage: star <rho_0> <C> <lambda> <dr>
```

and must produce the following (and only the following!) on standard output:

```
<r_1>   <rho_1>   <P_1>   <m_1>
<r_2>   <rho_2>   <P_2>   <m_2>
                    .
                    .
                    .
<r_n>   <rho_n>   <P_n>   <m_n>
```

i.e., there should be 4 numbers per line (with no extraneous characters such as "`r =`", commas, etc.). Note that

```
<r_1> = 0.0
```

and

```
<r_2> - <r_1> = <r_3> - <r_2> = ... = <r_n> - <r_n-1> = <dr>
```

where `<dr>` is the fourth command line argument. The particular value of `<n>` (i.e. the number of lines `star` spews out), will, in general, vary as a function of the input arguments: the outer edge of the star is to be defined as the maximum radius (to a precision `<dr>`) at which the pressure is still positive. Test your program thoroughly.

Use your program (or convert `star` to a subroutine, and embed it in another program of your own design), to perform "parameter space surveys" in the free parameters

$$\rho(0), \quad C, \quad \lambda$$

At a minimum, construct some plots of total mass vs. central density, and stellar radius vs. central density for several values of C and λ. Are there any values of C and/or λ which appear particularly interesting? Document your program(s) thoroughly (including testing procedures) and discuss your findings.