Physics 305 – Computational Physics
Term Project
Due Date: Friday, December 12, 5pm

The program in your term project can be written in either C/C++ or python, although C/C++ is the preferred language. The program, an explanation of what the program does, along with answers to **all** questions asked should be uploaded in the `Homeworks` directory on noether, where you will need to create a subdirectory called `TermProject`. For programs written in C/C++ you should submit **one** Makefile that compile the code using the gnu C++ compiler by simply typing `make`. Your Makefile should also have a `clean` target that removes executable and object files.

You are expected to write a term paper (in word or Latex) on your project that discusses the problem you are trying to solve, the basic equations that govern the problem, includes plots that show the solutions, and describes the solution and the numerical method involved. In addition, you must demonstrate that your solution is correct by showing that the code converges at the expected order. If your code does not converge at the expected order you should try to identify potential reasons for why this is the case. You are expected to work on your term project by yourself..

Your term project will receive full credit **only** if: (a) the program compile successfully using make, (python programs have to run without error using python 3), (b) the programs have explanatory comments and variable names that identify with the problem equations you are trying to solve, (c) the programs are properly indented, (d) give the correct output, and (e) demonstrate the validity of the solution through convergence plots. No credit will be given to late term projects.

The term paper is as important as the code (50% of the term project credit will go to the code and the other 50% to the paper). Answers to the questions and analysis requested below should be elaborated in the report. Plots should be clearly labeled and be properly described in the report, and not just shown. You will need to explain what each and every plot demonstrates. A polished paper written in word or LaTex is expexted to get full credit.

Note: Before you present results from numerical integrations that answer the questions in the project, it is critical to *first* perform the convergence tests for one case, and to estimate errors. This will tell you how small a step size is necessary for accurate solutions. Only after errors are estimated, does it make sense to run your code for producing results that help you learn more about the system you study.

I. SOLVING MAXWELL'S EQUATIONS IN 3D: A CIRCULAR-ORBIT DIPOLE SOURCE

This project guide describes how to solve Maxwell's equations in three spatial dimensions using a second–order, explicit finite-difference time-domain (FDTD/Yee) scheme. The electromagnetic source is a charge *dipole* consisting of two point charges $\pm q$ in circular orbit, producing a time-varying current and charge density.

II. GOVERNING EQUATIONS (SI UNITS)

We solve the Maxwell equations in vacuum with sources:

$$\mu_0 \frac{\partial \mathbf{H}}{\partial t} = -\nabla \times \mathbf{E}, \tag{1}$$

$$\varepsilon_0 \frac{\partial \mathbf{E}}{\partial t} = \nabla \times \mathbf{H} - \mathbf{J}, \tag{2}$$

supplemented by Gauss's laws

$$\nabla \cdot \mathbf{E} = \rho/\varepsilon_0, \qquad \nabla \cdot \mathbf{H} = 0. \tag{3}$$

III. CIRCULAR-ORBIT CHARGE-DIPOLE SOURCE

We model two point charges $\pm q$ orbiting in the x–y plane about the origin with radius a and angular frequency Ω. Their trajectories are

$$\mathbf{r}_{\pm}(t) = \pm a\,[\cos(\Omega t)\,\hat{\mathbf{x}} + \sin(\Omega t)\,\hat{\mathbf{y}}], \qquad \mathbf{v}_{\pm}(t) = \frac{\mathrm{d}\mathbf{r}_{\pm}}{\mathrm{d}t} = \pm a\Omega[-\sin(\Omega t)\,\hat{\mathbf{x}} + \cos(\Omega t)\,\hat{\mathbf{y}}]. \tag{4}$$

Notice that a sets the length-scale in this problem. We regularize the delta functions with a narrow Gaussian of width σ:

$$G_\sigma(\mathbf{x}) = \frac{1}{(\sqrt{\pi}\sigma)^3}\exp\!\left(-\frac{x^2+y^2+z^2}{\sigma^2}\right). \tag{5}$$

Charge and current densities are

$$\rho(\mathbf{x},t) = \; q\,G_\sigma(\mathbf{x}-\mathbf{r}_+(t)) \;-\; q\,G_\sigma(\mathbf{x}-\mathbf{r}_-(t))\,, \tag{6}$$

$$\mathbf{J}(\mathbf{x},t) = \; q\,\mathbf{v}_+(t)\,G_\sigma(\mathbf{x}-\mathbf{r}_+(t)) \;-\; q\,\mathbf{v}_-(t)\,G_\sigma(\mathbf{x}-\mathbf{r}_-(t))\,. \tag{7}$$

a. Dipole Moment and Frequency. $\mathbf{p}(t) = 2qa[\cos(\Omega t)\,\hat{\mathbf{x}} + \sin(\Omega t)\,\hat{\mathbf{y}}]$, with dominant radiation wavelength $\lambda \approx 2\pi c/\Omega$.

b. Nondimensionalization. Define reference length L, time $T = L/c$, fields $E_\star$ and $H_\star = E_\star/Z_0$ with free-space impedance $Z_0 = \sqrt{\mu_0/\varepsilon_0}$. Then all equations can be written in dimensionless form (set $c = \varepsilon_0 = \mu_0 = 1$). For implementation in code you may retain SI units and constants explicitly, if you want.

IV. COMPUTATIONAL GRID: YEE STAGGERING

We use the standard Yee grid, which staggers field components in space and time to achieve second-order accuracy and exact discrete curl identities. Let indices (i,j,k) denote cell centers with spacings $(\Delta x, \Delta y, \Delta z)$. The magnetic field is advanced at half time steps $t^{n+1/2} = (n+\frac{1}{2})\Delta t$, and the electric field at integer steps $t^n = n\Delta t$. Components are stored on faces/edges:

$$E_x\big|^{n}_{i+\frac{1}{2},j,k}, \quad E_y\big|^{n}_{i,j+\frac{1}{2},k}, \quad E_z\big|^{n}_{i,j,k+\frac{1}{2}},$$

$$H_x\big|^{n+\frac{1}{2}}_{i,j+\frac{1}{2},k+\frac{1}{2}}, \quad H_y\big|^{n+\frac{1}{2}}_{i+\frac{1}{2},j,k+\frac{1}{2}}, \quad H_z\big|^{n+\frac{1}{2}}_{i+\frac{1}{2},j+\frac{1}{2},k}.$$

The current density $\mathbf{J}$ is collocated with $\mathbf{E}$ (same component locations) at $t^{n+\frac{1}{2}}$ for a centered forcing in Ampère's law.

V. DISCRETE UPDATE EQUATIONS (VECTOR FORM)

Discretize the curls with centered second-order differences. The leapfrog scheme reads

$$\mathbf{H}^{\,n+\frac{1}{2}} = \mathbf{H}^{\,n-\frac{1}{2}} - \frac{\Delta t}{\mu_0}\,\nabla_h \times \mathbf{E}^{\,n}, \tag{8}$$

$$\mathbf{E}^{\,n+1} = \mathbf{E}^{\,n} + \frac{\Delta t}{\varepsilon_0}\left(\nabla_h \times \mathbf{H}^{\,n+\frac{1}{2}} - \mathbf{J}^{\,n+\frac{1}{2}}\right). \tag{9}$$

VI. FULL COMPONENT-WISE YEE UPDATES

For clarity, here are *all six* update equations expanded component-wise with proper line breaks. Throughout, omit indices on the right-hand side when obvious; every term is evaluated at the location of the left-hand side component.

Magnetic Field Updates (Faraday's Law)

$$H_x^{n+\frac{1}{2}}\Big|_{i,\,j+\frac{1}{2},\,k+\frac{1}{2}} = H_x^{n-\frac{1}{2}}\Big|_{i,\,j+\frac{1}{2},\,k+\frac{1}{2}} - \frac{\Delta t}{\mu_0}\left[\frac{E_z^n\big|_{i,\,j+1,\,k+\frac{1}{2}} - E_z^n\big|_{i,\,j,\,k+\frac{1}{2}}}{\Delta y} - \frac{E_y^n\big|_{i,\,j+\frac{1}{2},\,k+1} - E_y^n\big|_{i,\,j+\frac{1}{2},\,k}}{\Delta z}\right], \tag{10}$$

$$H_y^{n+\frac{1}{2}}\Big|_{i+\frac{1}{2},\,j,\,k+\frac{1}{2}} = H_y^{n-\frac{1}{2}}\Big|_{i+\frac{1}{2},\,j,\,k+\frac{1}{2}} - \frac{\Delta t}{\mu_0}\left[\frac{E_x^n\big|_{i+\frac{1}{2},\,j,\,k+1} - E_x^n\big|_{i+\frac{1}{2},\,j,\,k}}{\Delta z} - \frac{E_z^n\big|_{i+1,\,j,\,k+\frac{1}{2}} - E_z^n\big|_{i,\,j,\,k+\frac{1}{2}}}{\Delta x}\right], \tag{11}$$

$$H_z^{n+\frac{1}{2}}\Big|_{i+\frac{1}{2},\,j+\frac{1}{2},\,k} = H_z^{n-\frac{1}{2}}\Big|_{i+\frac{1}{2},\,j+\frac{1}{2},\,k} - \frac{\Delta t}{\mu_0}\left[\frac{E_y^n\big|_{i+1,\,j+\frac{1}{2},\,k} - E_y^n\big|_{i,\,j+\frac{1}{2},\,k}}{\Delta x} - \frac{E_x^n\big|_{i+\frac{1}{2},\,j+1,\,k} - E_x^n\big|_{i+\frac{1}{2},\,j,\,k}}{\Delta y}\right]. \tag{12}$$

Evaluate $\mathbf{J}$ at $t^{n+\frac{1}{2}}$ and sample ρ at t^n or t^{n+1}.

Electric Field Updates (Ampère–Maxwell Law)

$$E_x^{n+1}\Big|_{i+\frac{1}{2},\,j,\,k} = E_x^n\Big|_{i+\frac{1}{2},\,j,\,k} + \frac{\Delta t}{\varepsilon_0}\left[\frac{H_z^{n+\frac{1}{2}}\big|_{i+\frac{1}{2},\,j+\frac{1}{2},\,k} - H_z^{n+\frac{1}{2}}\big|_{i+\frac{1}{2},\,j-\frac{1}{2},\,k}}{\Delta y} - \frac{H_y^{n+\frac{1}{2}}\big|_{i+\frac{1}{2},\,j,\,k+\frac{1}{2}} - H_y^{n+\frac{1}{2}}\big|_{i+\frac{1}{2},\,j,\,k-\frac{1}{2}}}{\Delta z}\right] - \frac{\Delta t}{\varepsilon_0}\,J_x^{n+\frac{1}{2}}\Big|_{i+\frac{1}{2},\,j,\,k}, \tag{13}$$

$$E_y^{n+1}\Big|_{i,\,j+\frac{1}{2},\,k} = E_y^n\Big|_{i,\,j+\frac{1}{2},\,k} + \frac{\Delta t}{\varepsilon_0}\left[\frac{H_x^{n+\frac{1}{2}}\big|_{i,\,j+\frac{1}{2},\,k+\frac{1}{2}} - H_x^{n+\frac{1}{2}}\big|_{i,\,j+\frac{1}{2},\,k-\frac{1}{2}}}{\Delta z} - \frac{H_z^{n+\frac{1}{2}}\big|_{i+\frac{1}{2},\,j+\frac{1}{2},\,k} - H_z^{n+\frac{1}{2}}\big|_{i-\frac{1}{2},\,j+\frac{1}{2},\,k}}{\Delta x}\right] - \frac{\Delta t}{\varepsilon_0}\,J_y^{n+\frac{1}{2}}\Big|_{i,\,j+\frac{1}{2},\,k}, \tag{14}$$

$$E_z^{n+1}\Big|_{i,\,j,\,k+\frac{1}{2}} = E_z^n\Big|_{i,\,j,\,k+\frac{1}{2}} + \frac{\Delta t}{\varepsilon_0}\left[\frac{H_y^{n+\frac{1}{2}}\big|_{i+\frac{1}{2},\,j,\,k+\frac{1}{2}} - H_y^{n+\frac{1}{2}}\big|_{i-\frac{1}{2},\,j,\,k+\frac{1}{2}}}{\Delta x} - \frac{H_x^{n+\frac{1}{2}}\big|_{i,\,j+\frac{1}{2},\,k+\frac{1}{2}} - H_x^{n+\frac{1}{2}}\big|_{i,\,j-\frac{1}{2},\,k+\frac{1}{2}}}{\Delta y}\right] - \frac{\Delta t}{\varepsilon_0}\,J_z^{n+\frac{1}{2}}\Big|_{i,\,j,\,k+\frac{1}{2}}. \tag{15}$$

Discrete Divergence to monitor the Gauss constraint

Define the discrete divergence of $\mathbf{E}$ at cell centers:

$$(\nabla_h \cdot \mathbf{E})^n\Big|_{i,j,k} = \frac{E_x^n\big|_{i+\frac{1}{2},\,j,\,k} - E_x^n\big|_{i-\frac{1}{2},\,j,\,k}}{\Delta x} + \frac{E_y^n\big|_{i,\,j+\frac{1}{2},\,k} - E_y^n\big|_{i,\,j-\frac{1}{2},\,k}}{\Delta y} + \frac{E_z^n\big|_{i,\,j,\,k+\frac{1}{2}} - E_z^n\big|_{i,\,j,\,k-\frac{1}{2}}}{\Delta z}. \quad (16)$$

Then define the constrain $C_E = (\nabla_h \cdot \mathbf{E})^n\Big|_{i,j,k} - \rho^n_{i,j,k} = 0$. Similarly, you need a discretized version of the no-magnetic-monopole constraint $C_H = \nabla \cdot \mathbf{H} = 0$.

These constraints should be near 0 for the integration to be physically correct. You need to monitor these constraints for validity.

VII. CONSTRAINT-SATISFYING INITIALIZATION AND RELAXATION SCHEME

The leapfrog Yee scheme preserves Gauss's laws if they hold at initialization and if sources obey discrete continuity. This section provides a practical, grid-consistent initialization for $\mathbf{E}^0$ and $\mathbf{H}^{1/2}$ and an iterative relaxation method to solve the Poisson equation used by the initialization.

A. Constraint-Satisfying Initial Data

Given a discrete charge density ρ^0 (e.g., the Gaussian-regularized dipole at $t = 0$), enforce Gauss–E

$$\nabla_h \cdot \mathbf{E}^0 = \rho^0/\varepsilon_0 \quad (17)$$

by solving the discrete Poisson equation for a scalar potential ϕ:

$$\nabla_h^2 \phi = -\rho^0/\varepsilon_0, \qquad \mathbf{E}^0 = -\nabla_h \phi. \quad (18)$$

Initialize the magnetic field at half time level with zero,

$$\mathbf{H}^{1/2} = \mathbf{0}, \quad (19)$$

which satisfies $\nabla_h \cdot \mathbf{B}^{1/2} = 0$ and is compatible with the first Faraday update.

a. Boundary conditions for ϕ. Choose BCs consistent with your outer treatment:

- **Dirichlet** $\phi = 0$ on the box faces (good "open" approximation if the source is far from boundaries).

B. Relaxation Schemes for the Poisson Solve

We sketch three classic iterative relaxations to solve $\nabla_h^2 \phi = -\rho^0/\varepsilon_0$ at $t = 0$. Let $h_x = \Delta x$, $h_y = \Delta y$, $h_z = \Delta z$, and define

$$\alpha = \frac{1}{\frac{2}{h_x^2} + \frac{2}{h_y^2} + \frac{2}{h_z^2}}. \quad (20)$$

Denote the right-hand side $f_{i,j,k} = -\rho^0_{i,j,k}/\varepsilon_0$.

a. (1) Jacobi. Update all nodes from the previous sweep:

$$\phi^{(m+1)}_{i,j,k} = \alpha\Big[\frac{\phi^{(m)}_{i+1,j,k} + \phi^{(m)}_{i-1,j,k}}{h_x^2} + \frac{\phi^{(m)}_{i,j+1,k} + \phi^{(m)}_{i,j-1,k}}{h_y^2} + \frac{\phi^{(m)}_{i,j,k+1} + \phi^{(m)}_{i,j,k-1}}{h_z^2} - f_{i,j,k}\Big]. \quad (21)$$

b. (2) Gauss–Seidel Use newest values as soon as they are available:

$$\phi_{i,j,k}^{(m+1)} = \alpha\Big[\frac{\phi_{i+1,j,k}^{(m+1)} + \phi_{i-1,j,k}^{(m+1)}}{h_x^2} + \frac{\phi_{i,j+1,k}^{(m+1)} + \phi_{i,j-1,k}^{(m+1)}}{h_y^2} + \frac{\phi_{i,j,k+1}^{(m+1)} + \phi_{i,j,k-1}^{(m+1)}}{h_z^2} - f_{i,j,k}\Big]. \tag{22}$$

(Implement in a fixed sweep order or in red–black ordering for parallelism.)

c. (3) Successive Over-Relaxation (SOR). Accelerate Gauss–Seidel with a relaxation factor $\omega \in (1, 2)$:

$$\phi_{i,j,k}^{(m+1)} = (1-\omega)\,\phi_{i,j,k}^{(m)} + \omega\,\phi_{i,j,k}^{\mathrm{GS}}, \tag{23}$$

where $\phi_{i,j,k}^{\mathrm{GS}}$ is the Gauss–Seidel update. For a cubic grid, a good starting choice is $\omega \approx 1.8$; tune for fastest decay of the residual.

d. Residual and stopping criterion. At sweep m compute

$$r_{i,j,k}^{(m)} = f_{i,j,k} - (\nabla_h^2 \phi^{(m)})_{i,j,k}, \qquad \|r^{(m)}\|_2 = \Big(\sum_{i,j,k} (r_{i,j,k}^{(m)})^2 \Delta V\Big)^{1/2}. \tag{24}$$

Stop when $\|r^{(m)}\|_2 / \|f\|_2 < \varepsilon$ (e.g., 10^{-8}) or when the maximum componentwise residual falls below a tolerance. After convergence, set $\mathbf{E}^0 = -\nabla_h \phi$, using the discrete operators below.

e. Notes. Jacobi is simple but slow; Gauss–Seidel speeds up convergence; SOR can reduce iterations by an order of magnitude. For large domains, one considers a multigrid V-cycle with Gauss–Seidel smoothing. For this project use, the Gauss–Seidel and SOR techniques.

C. Discrete Operators (Yee-compatible)

At cell centers (i, j, k),

$$(\nabla_h \cdot \mathbf{E})_{i,j,k} = \frac{E_x|_{i+\frac{1}{2},j,k} - E_x|_{i-\frac{1}{2},j,k}}{\Delta x} + \frac{E_y|_{i,j+\frac{1}{2},k} - E_y|_{i,j-\frac{1}{2},k}}{\Delta y} + \frac{E_z|_{i,j,k+\frac{1}{2}} - E_z|_{i,j,k-\frac{1}{2}}}{\Delta z}, \tag{25}$$

$$(\nabla_h^2 \phi)_{i,j,k} = \frac{\phi_{i+1,j,k} - 2\phi_{i,j,k} + \phi_{i-1,j,k}}{\Delta x^2} + \frac{\phi_{i,j+1,k} - 2\phi_{i,j,k} + \phi_{i,j-1,k}}{\Delta y^2} + \frac{\phi_{i,j,k+1} - 2\phi_{i,j,k} + \phi_{i,j,k-1}}{\Delta z^2}. \tag{26}$$

Face-centered components of $\nabla_h \phi$ are obtained with centered differences, e.g.

$$E_x|_{i+\frac{1}{2},j,k} = -\frac{\phi_{i+1,j,k} - \phi_{i,j,k}}{\Delta x}, \quad E_y|_{i,j+\frac{1}{2},k} = -\frac{\phi_{i,j+1,k} - \phi_{i,j,k}}{\Delta y}, \quad E_z|_{i,j,k+\frac{1}{2}} = -\frac{\phi_{i,j,k+1} - \phi_{i,j,k}}{\Delta z}. \tag{27}$$

VIII. BOUNDARY CONDITIONS "FAR AWAY" FOR THE EVOLUTION

Place boundaries several wavelengths from the source and use absorbing boundary conditions (ABCs).

First-Order Mur ABC on Cartesian Box Faces

Let u denote any tangential field component at a boundary face. For the $x = x_{\max}$ face (outward normal $+\hat{\mathbf{x}}$), with grid index $i = i_{\max}$ and interior neighbor $i_{\max} - 1$,

$$u^{n+1}\Big|_{i_{\max},\,j,\,k} = u^{n}\Big|_{i_{\max}-1,\,j,\,k} + \frac{c\Delta t - \Delta x}{c\Delta t + \Delta x}\Big[u^{n+1}\Big|_{i_{\max}-1,\,j,\,k} - u^{n}\Big|_{i_{\max},\,j,\,k}\Big]. \tag{28}$$

Similarly, for the $x = x_{\min}$ face (outward normal $-\hat{\mathbf{x}}$),

$$u^{n+1}\Big|_{i_{\min},\,j,\,k} = u^{n}\Big|_{i_{\min}+1,\,j,\,k} + \frac{c\Delta t - \Delta x}{c\Delta t + \Delta x}\Big[u^{n+1}\Big|_{i_{\min}+1,\,j,\,k} - u^{n}\Big|_{i_{\min},\,j,\,k}\Big]. \tag{29}$$

Analogous formulas hold for the y- and z-faces by replacing $\Delta x \to \Delta y$ (or Δz) and shifting the corresponding indices $(j \pm 1)$ or $(k \pm 1)$. Apply Mur to each tangential component: at x-faces, update E_y, E_z, H_y, H_z; at y-faces update E_x, E_z, H_x, H_z; and at z-faces update E_x, E_y, H_x, H_y.

IX. CFL STABILITY CONDITION FOR THE EVOLUTION

For a uniform Cartesian grid the explicit scheme is stable if

$$\Delta t \leq \frac{1}{c}\left(\frac{1}{\Delta x^2}+\frac{1}{\Delta y^2}+\frac{1}{\Delta z^2}\right)^{-1/2}. \tag{30}$$

Since you will be using $\Delta x = \Delta y = \Delta z$ for this project the stability condition becomes

$$\frac{c\sqrt{3}\Delta t}{\Delta x} \leq 1. \tag{31}$$

X. ALGORITHM (STEP-BY-STEP)

1. **Initialize:** Construct a 3-dimensional grid with domain size several wavelength. Set $\mathbf{E}^0$ using the relaxation scheme described above, and $\mathbf{H}^{1/2} = \mathbf{0}$. Choose Δt from CFL. Initialize ρ^0 and $\mathbf{J}^{1/2}$ from the dipole.
2. **Loop over n:**
   (a) Build $\mathbf{J}^{n+\frac{1}{2}}$ (and ρ^{n+1} from the analytic sources provided, after you have chosen charge q and frequency Ω.
   (b) Update $\mathbf{H}^{n+\frac{1}{2}}$ via (10)–(12).
   (c) Update $\mathbf{E}^{n+1}$ via (13)–(15).
   (d) Apply BCs on all faces.
   (e) Monitor Gauss law error and energy.

XI. DIAGNOSTICS & VALIDATION

1. Compute the total Energy $U^n = \sum(\frac{1}{2}\varepsilon_0|\mathbf{E}|^2 + \frac{1}{2}\mu_0|\mathbf{H}|^2)\Delta V$ vs time
2. Compute Poynting flux $\mathbf{S} = \mathbf{E} \times \mathbf{H}$ through a box far away
3. Monitor the discrete Gauss law residual
4. In the far field verify dipole-like angular pattern and $E/H \approx Z_0$.
5. Perform a resolution study and demonstrate that Gauss' constraints converge to 0 with increasing resolution.
6. **Convergence**: Perform a self-convergence study by running your code with 3 or more resolutions. Focus on a single point in space and time for the resolution study. Does the order of convergence match your expectation? If not try to explain why this is case.
7. Using the order of convergence you determined, employ Richardson extrapolation to determine an error for the solution of the equations..