**Functional Features**

**1. Priority Levels**

* Allow users to set priority: High, Medium, Low.
* You can use color codes or symbols to differentiate them.

**2. Due Dates / Deadlines**

* Allow users to add due dates to tasks.
* Use <ctime> for date/time operations.
* Show how many days are left or if the task is overdue.

**3. Categories or Tags**

* Group tasks under categories (e.g., Work, Personal, Study).
* Allow filtering based on category.

**4. Status Tracking**

* Mark tasks as: Not Started, In Progress, Done.
* Use symbols or different text formats to indicate status.

**5. Save to File (Persistence)**

* Save/load task list to/from a file (.txt, .json, etc.)
* Use fstream for file handling.

**6. Search and Filter**

* Search by keyword, category, or due date.
* Filter by priority or completion status.

**7. Sorting Options**

* Sort tasks by deadline, priority, status, etc.

**8. Recurring Tasks**

* Add support for daily/weekly recurring tasks.

**9. Task Descriptions**

* Support multi-line task details beyond just the title.

**10. Reminders or Alerts**

* Optional: use system notifications or terminal-based reminders.

**🎨 Visual Improvements (Console UI)**

**1. Color Output**

* Use ANSI escape codes for colored text:
* #define RED "\033[31m"
* #define GREEN "\033[32m"
* #define YELLOW "\033[33m"
* #define RESET "\033[0m"
* std::cout << RED << "High Priority Task" << RESET << std::endl;

**2. Better Formatting**

* Align columns using setw() from <iomanip>:
* std::cout << std::left << std::setw(20) << "Task"
* << std::setw(10) << "Priority"
* << std::setw(15) << "Due Date" << std::endl;

**3. Boxed UI / Menus**

* Create a basic ASCII UI with borders, menus, etc.
* +------------------------------------+
* | TO-DO LIST MENU |
* +------------------------------------+
* | 1. Add Task |
* | 2. View Tasks |
* | 3. Delete Task |
* | 4. Save and Exit |
* +------------------------------------+

**4. Progress Bar / Task Count**

* Show total tasks, completed tasks, percentage done.
* [#####-----] 50% Complete

**5. Animation or Effects (Optional)**

* Simulate typing or loading effects for better UX.

**💡 Bonus Ideas**

* **Password Protection**: Simple login for accessing task list.
* **Export to CSV or HTML**: Useful for sharing.
* **Undo/Delete Recovery**: Let users undo last delete action.
* **Use of Classes & OOP**: Make your code modular and easier to maintain.

Would you like a **sample C++ code** showing some of these features (like colors, priorities, or saving to file)?

View tasks

* View completed tasks
* View all tasks
* View tasks according to priority
* View tasks based on category
* View tasks based on due date