# Understanding the C++ Program for Searching and Calculating Percentage in Excel (Using OpenXLSX)

**1. Basic C++ Concepts**

**Input & Output:**

* cin is used to take user input.
* cout is used to display output on the console.

**Variables & Data Types:**

* string for storing PIN numbers.
* int and double for numerical calculations.

**Control Structures:**

* if-else is used to check conditions.
* for and while loops are used for iteration.

**Functions:**

* trim() function removes extra spaces.
* toLower() converts strings to lowercase for case-insensitive search.

**Exception Handling:**

* try-catch is used to handle file errors.

**2. File Handling in C++**

* Files are accessed in C++ using streams.
* The OpenXLSX library is used to interact with Excel files.

**3. String Manipulation**

* trim() removes leading and trailing spaces.
* toLower() ensures case-insensitive comparison.
* find\_if() is used to detect the first non-space character.
* erase() is used to remove unwanted characters.
* transform() is used to convert strings to lowercase.

**4. Working with Excel Files using OpenXLSX**

**Opening an Excel File:**

XLDocument doc;

doc.open(filePath);

**Reading Worksheets:**

auto workbook = doc.workbook();

XLWorksheet sheet = workbook.worksheet(sheetName);

**Reading Cells:**

XLCellValue cellValue = sheet.cell(row, col).value();

**Handling Different Data Types:**

string pinValue = cellValue.get<string>();

int marks = cellValue.get<int64\_t>();

double percentage = cellValue.get<double>();

**5. Searching & Data Filtering**

* The program loops through the rows of an Excel sheet to find a matching PIN number.
* It compares searchPIN with values in the "Pin Number" column.

if (pinValue == searchPIN) {

cout << "Details found in sheet: " << sheetName << endl;

}

**6. Calculating Total Marks & Percentage**

**Summing Marks:**

int totalMarks = 0;

for (size\_t col = 3; col <= colCount; ++col) {

totalMarks += sheet.cell(row, col).value().get<int64\_t>();

}

**Calculating Percentage:**

double percentage = (static\_cast<double>(totalMarks) / totalPossibleMarks) \* 100;

cout << "Percentage: " << percentage << "%" << endl;

**7. Error Handling & Debugging**

**Handling File Errors:**

try {

doc.open(filePath);

} catch (const exception& e) {

cerr << "Error: " << e.what() << endl;

}

**Handling Incorrect Data Types:**

* Checking XLValueType before accessing cell values prevents errors.

**8. Compilation & Running the Program**

**Compiling with OpenXLSX:**

g++ search.cpp -o search -I"path/to/OpenXLSX" -L"path/to/lib" -lOpenXLSX

**Running the Program:**

./search

**9. Extending the Program**

**Search by Student Name:**

* Modify the search condition to check the "Name of the Student" column.

**Exporting Output to Excel:**

* Use doc.save() to save new results.

**Adding a GUI (Optional):**

* Use Qt for a graphical interface.

**Conclusion**

This program demonstrates how to read, search, and calculate data from an Excel file using **OpenXLSX** in C++. Understanding file handling, string manipulation, and error handling is crucial to extending this project.