## 4.pseduocode for Academic performance analyzer

**Step 1:** Welcome message and number of students:

Step1.1: Print a welcome message and a line.

Step 1.2: Read the number of students from the user and store it in the variable numstudents.

**Step 2:** Student data entry:

Loop numstudents times:

Step 2.1: Ask the user for the name of the student and store it in the names array at the current index.

Step 2.2: Generate an ID for the student by concatenating “ets” with the current index + 1 and store it in the id array at the current index.

Loop 7 times:

Step 2.3: Ask the user for the student’s hours spent studying on each day of the week and store them in the hours array at the current student and day indices.

**Step 3:** Calculate total hours:

Step 3.1: Set a variable total to 0.

Loop 7 times:

Step 3.2: Add the student’s hours for the current day to the total variable.

Step 3.3: Store the total hours for the student in the totals array at the current index.

**Step 4:** Print results table:

Step 4.1: Print a line stating “Results in tabular format:”.

Step 4.2: Print a header row with “Student Name” and “Day 1” to “Day 7” and “Total” columns.

Loop numstudents times:

Step 4.3: Print the student’s name and ID.

Loop 7 times:

Step 4.4: Print the student’s hours for the current day.

Step 4.5: Print the student’s total hours.

**Step 5:** Search for a student:

Step 5.1: Print a line asking the user to search by name or ID.

Step 5.2: Get the user’s search query and store it in the variable search.

Step 5.3: Convert the search query to lowercase.

Loop numstudents times:

Step 5.4: Convert the current student's name and ID to lowercase.

Step 5.6: Check if the search query matches the current student's name or ID.

Step 5.7: If a match is found, set found to the current student index and break the loop.

**Step 6**: Display search results:

Step 6.1: If no student was found, print a message stating "No student found with that name or ID".

Step 6.2: If a student was found, Print a line with a separator.

Step 6.3: Print a message stating "Found user named [student name] ([student ID])".

Step 6.4: Print another line with a separator.

Step 6.5: Print a header row with "Student Name" and "Day 1" to "Day 7" and "Total" columns.

Step 6.6: Print the found student's name, ID, hours for each day, and total hours.

**Step 7:** Ask to search again:

Step 7.1: If a student was found, ask the user if they want to search again.

Step 7.2: If the user chooses to search again, go back to step 5.

Step 7.3: If the user chooses not to search again, print a thank you message.

**Step 8:** Exit program:

Step 8.1: Return 0 to exit the program.

## Section A Group 4

| | |
|---|---|
| Betselot Getahun | ETS0277/15 |
| Bewnet Adisalem | ETS0280/15 |
| Bezawit Taye | ETS0284/15 |
| Bilen Manaye | ETS0290/15 |
| Biniyam Abebe | ETS 0294/15 |
| Biniyam Cheru | ETS0296/15 |
| Biniyam Yalew | ETS0297/15 |
| Biruk Mitiku | ETS0302/15 |
| Bitsuan Abate | ETS0328/15 |
| Dagim Tadesse | ETS0343/15 |
| Dagim Abraham | ETS0344/15 |