AGTE 411: Precision Agriculture Assignment

- Paul Nugent (Precision Agriculture)

Purpose: This assignment aims to introduce students to the concepts of pseudocode and its application in programming. By leveraging generative AI tools, such as Google Gemini, students will gain hands-on experience in self-guided learning, evaluation of AI-generated content, and the iterative development of pseudocode. This assignment aims to help students build confidence in integrating AI technologies to overcome programming hurdles. This process will be used throughout the semester as we develop the shared codebase to implement our Internet of Things (IoT) sensor network.

Duration: 2 hours

Learning Objectives

By the end of this assignment, students will be able to:

1. Apply generative AI tools to solve a discipline-specific problem
2. Critically evaluate AI-generated content for accuracy, bias, and relevance
3. Apply AI generative content in self-guided learning
4. Develop pseudocode that can be systematically converted into functional Arduino code using AI tools
5. Demonstrate understanding of the programming logic and hardware interactions on the Arduino platform.

Materials Needed

- Arduino MRK Wifi 1010
- HMP35C Temperature and Humidity Sensor
- Relay Module (5V)
- 12V DC Fan
- LED Module with integrated 220-ohm current limiting resistor
- Connector Wires
- 12V DC Fan

Generative AI Tool for Assignment

See the attached model card for more in-depth information on this AI tool.

- Name of tool: Gemini 2.0 Advanced
- Purpose: Conversational AI designed to assist with generating text-based outputs
- Who created it/Version used: Developed by Google, 2.0 Advanced
- Any known limitations/biases:
    - Knowledge cutoff of October 2023.
    - responses are based on patterns in training data, which may include cultural or systemic biases present in the data.

- performance depends on the clarity and detail of user inputs

Assignment Structure: Introduction to Arduino and Pseudocode in Agricultural IoT Systems

This assignment is designed to introduce you to the Arduino platform and the concept of pseudocode, key components we will use this semester to build our Internet of Things (IoT) sensor network.

Objective: Students will explicitly use Google Gemini to guide their process of pseudocode development.

1. You will use the Google Gemini Large Language Model (LLM) as a learning tool to understand how to write pseudocode and apply this knowledge to develop a system that measures and responds to temperature and humidity.
2. In class, your pseudocode will be reviewed by the instructor, shared with peers for discussion, and collaboratively converted into functional Arduino code using an AI LLM. This assignment lays the groundwork for future projects where you will create more advanced IoT systems.

Ethical Considerations: The assignment demonstrates how students can use LLMs like Google Gemini to self-learn and tackle tasks beyond their comfort zone. However, the impact of generative AI on learning depends on its use. At the same time, it can enhance understanding when used thoughtfully; over-reliance on AI-generated content without active engagement risks academic dishonesty and undermines genuine learning.

To ensure the ethical and beneficial use of generative AI:

- Within the AGTE 411 class, the use of AI is acceptable and even expected for these types of tasks, but this is only because it is stated within the assignment definition. This expectation will be established early in the class.

- Student are required to disclose how AI was used, including a full history of their prompts.

- This process will ensure that assignments promote genuine understanding and skill development by requiring systems to assess and reflect on the AI outputs

Pre-Reflection:

Review the full assignment and answer the following topics in your **Analysis and Reflection** document.

1. **AI Experience:** Have you used AI tool (ChatGPT, Gemini, Claude) in the past? If so, describe your experience with these AI tools.
2. **Programming Comfort:** On a scale of 1-10 how comfortable are you with computer programming. What challenges do you anticipate using programing to accomplish tasks with the Arduino systems?
3. **Learning with AI:** How do you feel about relying on an AI tool to teach you a new skill? What concerns or expectations do you have about using AI for this purpose?

Pseudocode Exploration:

1. In this section, you will use Google Gemini to learn how to write out the flow of a computer program in common English through pseudocode. This process will start with the following standard prompt

   "Explain pseudocode and how it's used to design programs for Arduino. Provide beginner-friendly examples that involve reading data from analog inputs (like a temperature sensor) and controlling digital outputs (like turning an LED on or off)."

2. After Gemini has responded to this prompt, follow up with at least four additional prompts asking for further information, examples, or use cases of pseudocode. You will define these prompts. Your goal is to learn about pseudocode, how it is used, and how it can be written. Document this exploration in your **Process Documentation** document that will be turned in on D2L.

3. Once you feel confident in your understanding of pseudocode, request Gemini to give you scenarios and test your ability to write the code with the following prompt

   "Give me a beginner-friendly Arduino scenario that requires using pseudocode with both analog inputs and digital outputs. I'll write the pseudocode, and you evaluate it."

   Work through at least three of these scenarios and save the output in your **Process Documentation** document. You are welcome to work through more, but you are only required to submit three examples.

Pseudocode Generation:

Once you are comfortable writing pseudocode, you will write pseudocode that defines the lab software required to complete the lab assignment for lab this week. In Lab 3, you will use the Arduino MKR Wifi 1010, connected to the HMPC25C temperature and humidity sensor (from the previous labs), to control a fan. The fan will turn ON when the temperature exceeds 30°C and turn OFF if it falls below 29°C.

Setting up Gemini to help you with this process will require three prompts. At first, we will tell it what we are doing and how we want it to respond, but ask it to wait for future input:

"I will describe an Arduino-based system to you. Help me write a clear and well-structured pseudocode. Do not provide any information until you have received the system definition and prompted to respond"

The second defines the system:

"This system uses an Arduino MKR WiFi 1010, an HMP35C sensor (0-1V calibrated output), an LED, a relay, and a 12V fan. The system calculates a 1-minute average temperature from the HMP35C sensor to control the fan: the fan turns ON if the temperature exceeds 30°C and OFF if it falls below 29°C."

The third prompts the system to request additional information that is required to assist you in writing pseudocode. NOTE: there is a high probability that prompt number three is not needed as Gemini will launch into the conversation on its own, but if needed use the following.

"Help me outline pin assignments and any special considerations for integrating these components before we write pseudocode for this task"

Now that you have set the stage, complete the following. Be sure to document this process in your **Process Documentation** file and submit the resulting code in your **Pseudocode** text file, both which will be submitted on D2L.

1. With the assistance of Google Gemini, write pseudocode to accomplish the tasks described above. Work through any questions that arise along the way.
2. Refine your pseudocode based on feedback from Google Gemini. Ask Google Gemini to evaluate your code and identify any weaknesses or areas that could be improved.
3. Test the clarity of their pseudocode by simulating a conversion to operational code.

Document the entire process, including:

1. Prompts Used: Record the exact prompts students input into Google Gemini.
2. AI Responses: Save Google Gemini's replies for reference.
3. Modifications: Document how the prompts evolved to refine pseudocode.

Reflection and Analysis:

Analyze the feedback provided by Google Gemini, from your analysis answer the following questions and add them to you **Analysis and Reflection** document.:

1. **AI Understanding of Prompts**: How accurately did Google Gemini interpret and respond to your initial prompts? Were there any instances where the AI misunderstood or required additional clarification? Provide examples to illustrate your observations.
2. **Clarity and Relevance of Responses**: Were the AI-generated responses clear and directly applicable to your task? Share specific examples of responses you found either helpful or confusing, and explain why.
3. **Pseudocode Refinement**: How effective was Google Gemini in helping you understand and refine your pseudocode? Compare your initial pseudocode to the final version and describe specific improvements suggested by the AI.

After completing the task, reflect on your experience by answering the following questions:

4. **Post-Task Reflection**: How do you feel about the process now that you've completed the task? What aspects of the assignment contributed to your sense of accomplishment or frustration?
5. **Comfort with AI-Driven Learning:** Compare your initial comfort level with using AI as a learning tool to how you feel now. How has this experience influenced your willingness to use AI in future learning?
6. **Evaluation of AI Usage**: What did you find most helpful about using Google Gemini during this task, and why? Conversely, what aspects were least helpful, and how could these be improved in future assignments?

7. **Overall Experience:** Reflecting on the entire process, how did using Google Gemini impact your learning experience? What did you find most helpful, and what aspects could have been improved?

In Lab Work:

1. In lab this week we will compare your pseudocode with the pseudocode of your peers. This will result in a finalized and shared version of the pseudocode from which we will use a generative AI tool to create a shared Arduino code that will become the foundation of our Arduino code base. See the lab handout for additional information.

Submission Guidelines:

Prior to lab, students will submit the following three documents to D2L:

- **Pseudocode:** [Text (.txt) document.] A well-organized and functional pseudocode that accomplishes the task set out in this assignment.
- **Process Documentation:** [MS-Word (.docx) document.] A summary of your interactions with Google Gemini, including prompts, responses, and modifications.
- **Analysis and Reflection:** [MS-Word (.docx) document.] A written response to the analysis and reflection questions (including pre-reflection questions).

Assessment Criteria

1. Clarity and Accuracy of the Pseudocode
    - Pseudocode is logically structured and follows a clear, step-by-step process.
    - System functionality is described accurately (e.g., correct pin assignments, logical flow).
2. Reflection
    - Student demonstrates thoughtful engagement with the task.
    - Responses to the reflection questions are clear and insightful, indicating an understanding of the assignment's technical and experiential aspects.

3. Rubric: Arduino and Pseudocode Assignment

15 Points Total

Criteria	Excellent (5 points)	Proficient (4 points)	Basic (3 points)	Needs Improvement (2 points)
Pseudocode Clarity	Pseudocode is logically structured, easy to follow, and accurately represents functionality.	Pseudocode is clear and mostly accurate, with minor issues in structure or functionality.	Pseudocode has significant issues in clarity or functionality but demonstrates some understanding.	Pseudocode lacks clarity, structure, and functionality, demonstrating minimal understanding.
Effective AI Usage	AI tools were used effectively to generate, refine, and enhance the pseudocode, with clear documentation.	AI tools were used well, with some documentation of interactions.	AI tools were used minimally or with limited documentation.	Little to no evidence of effective AI tool usage or documentation.
Reflection & Analysis	Reflection is insightful and detailed and demonstrates critical thinking about the process and learning outcomes.	Reflection is thoughtful, with some critical analysis and clear insights.	Reflection lacks depth but covers basic aspects of the task.	Reflection is minimal or missing, with little evidence of engagement.

Additional Resources

- Gemini by Google: Gemini
- Other LLM's: Chat GPT by OpenAI, Claude
- Arduino MRK WiFi 1010 MKR WiFi 1010 | Arduino Documentation
- HMP45C hmp45c.pdf
- Arduino language documentation Language Reference | Arduino Documentation