```python
# 1. MEMORY CLASS
class Memory:
    def __init__(self):
        self.data = []

    def store(self, item):
        self.data.append(item)

    def retrieve(self):
        return self.data

    def clear(self):
        self.data = []
# 2. TOOL CLASSES

class Tool:
    def use(self, text):
        raise NotImplementedError("Each tool must implement use().")


class SearchTool(Tool):
    def use(self, text):
        return f"[SearchTool] Searching for: {text}"


class SummaryTool(Tool):
    def use(self, text):
        return f"[SummaryTool] Summary: {text[:30]}..."


# 3. AGENT CLASS

class Agent:
    def __init__(self, name, role, memory, tools):
        self.name = name
        self.role = role
        self.memory = memory
        self.tools = tools

    def think(self, text):
        thought = f"{self.name} is thinking about: {text}"
        self.memory.store(thought)
        return thought

    def act(self, text, tool_name):
        if tool_name not in self.tools:
            return f"Tool '{tool_name}' not found."

        tool = self.tools[tool_name]
        result = tool.use(text)
        self.memory.store(result)
        return result

    def run(self, text):
        t = self.think(text)
        a = self.act(text, "search")
        return t, a


# Example Usage (You can remove this for assignment submission)
if __name__ == "__main__":
    memory = Memory()
    tools = {
        "search": SearchTool(),
        "summary": SummaryTool()
    }

    agent = Agent("AgentX", "Helper", memory, tools)

    print(agent.run("Python OOP assignment"))
    print(agent.act("This is an example paragraph for summary.", "summary"))
    print("Memory:", agent.memory.retrieve())
```