Sources used in the report

gosugamers.net
Beginner's guide to Marvel Snap: best starting decks, new cards and more | GosuGamers
Opens in a new window

bluestacks.com
Beginner's Guide for MARVEL Snap - Everything You Need to Know About This New CCG
Opens in a new window

developer.apple.com
Behind the Design: MARVEL SNAP - Discover - Apple Developer
Opens in a new window

m.youtube.com
8 mobile game design tips from the creators of MARVEL SNAP - YouTube
Opens in a new window

youtube.com
Marvel Snap Explained - YouTube
Opens in a new window

marvelsnapzone.com
Marvel Snap Beginner's Guide: How to Play, Tips and Tricks
Opens in a new window

gamerant.com
12 Beginner Tips For Marvel Snap - Game Rant
Opens in a new window

androidpolice.com
Marvel Snap beginner's guide: Tips and tricks for maintaining your win streak
Opens in a new window

marvelsnap.helpshift.com
How many types of cards are there? — MARVEL SNAP Help Center
Opens in a new window

reddit.com
Why some of the cards abilities are not what canon of the character would suggest (explained) : r/MarvelSnap - Reddit
Opens in a new window

reddit.com
An analysis of Marvel Snap Cards abilities vs Character Powers (and predictions) - Reddit
Opens in a new window

en.wikipedia.org
Marvel Snap - Wikipedia
Opens in a new window

marvelsnapzone.com
Marvel Snap Abilities: Full List and How They Work - Marvel Snap ...
Opens in a new window

blog.snap.untapped.gg
Snapping & Retreating in Marvel SNAP
Opens in a new window

reddit.com
A Beginners Guide to Marvel Snap : r/MarvelSnap - Reddit
Opens in a new window

techradar.com
I've played Marvel Snap every day since it came out, and these are the best Marvel Snap beginner decks I'd recommend right now | TechRadar
Opens in a new window

androidpolice.com
10 Marvel Snap tips and tricks for advanced players - Android Police
Opens in a new window

reddit.com
How 2 Snap, an exhaustive guide on the games central mechanic : r/MarvelSnap - Reddit
Opens in a new window

marvelsnapzone.com
Marvel Snap Series 1/2/3 Guide: Card Tier List & Beginner Decks
Opens in a new window

snap.cardsrealm.com
Marvel Snap: A Guide for Priority - Cards Realm
Opens in a new window

youtube.com
Starter Guide & Best early Decks for Marvel SNAP in 2025 - YouTube
Opens in a new window

youtube.com
Best Series One Decks | Marvel Snap Beginners Guide - YouTube
Opens in a new window

reddit.com
Beginner Deck Suggestions : r/MarvelSnapDecks - Reddit
Opens in a new window

bossrush.net
How To Play Marvel Snap: A Beginner's Guide - Boss Rush Network
Opens in a new window

deconstructoroffun.com
May 24 Marvel Snap - The Definitive Deconstruction - Deconstructor of Fun
Opens in a new window

leriohub.com
A Three Slots Resource Allocation Study Definitely Not In Marvel Snap | leriohub
Opens in a new window

marvelsnapzone.com
Starter Card Marvel Snap Cards - Marvel Snap Zone
Opens in a new window

ign.com
Pool 1 Card List - Marvel Snap Guide - IGN
Opens in a new window

ign.com
Tips and Strategies - Marvel Snap Guide - IGN
Opens in a new window

proceedings.neurips.cc
Opponent Modeling with In-context Search
Opens in a new window

reddit.com
Is MarvelSnap a strategy game? - Reddit
Opens in a new window

en.wikipedia.org
Monte Carlo tree search - Wikipedia
Opens in a new window

gameanalytics.com
Marvel Snap, Marvel Contest, CoD Mobile: What makes them a mass market hit? - GameAnalytics
Opens in a new window

snap.fan
Cards from the Starter collection - Marvel Snap - snap.fan
Opens in a new window

cnet.com
Marvel Snap: Beginner's Guide and Top Tips to Get Cards and Win ...
Opens in a new window

snap.fan
A Guide on Priority - snap.fan
Opens in a new window

snap.fan
Marvel Snap Cards - snap.fan
Opens in a new window

marvelsnapzone.com
Series 1 Beginner Decks - July 2025 - Marvel Snap Zone
Opens in a new window

arxiv.org
Magic: The Gathering is Turing Complete - arXiv
Opens in a new window

jair.org
A Survey on Opponent Modeling in Adversarial Domains - Journal of Artificial Intelligence Research
Opens in a new window

cs.cmu.edu
Game Theory-Based Opponent Modeling in Large Imperfect-Information Games - CMU School of Computer Science
Opens in a new window

core.ac.uk
Monte Carlo Tree Search for games with Hidden Information and Uncertainty - CORE
Opens in a new window

cs.mun.ca
Comparison of Monte Carlo Tree Search Methods in the Imperfect Information Card Game Cribbage - Computer Science
Opens in a new window

skatgame.net
Recursive Monte Carlo Search for Imperfect Information Games - Skat
Opens in a new window

marvelsnapzone.com
Marvel Snap Card Collection
Opens in a new window

marvelsnapzone.com
Marvel Snap Card Database
Opens in a new window

gamerant.com
Marvel Snap: Best Pool 2 Decks - Game Rant
Opens in a new window

ign.com
Marvel Snap Locations - IGN
Opens in a new window

marvelsnapzone.com
Series (Pool) 2 Beginner Decks - July 2025 - Marvel Snap Zone
Opens in a new window

gamerant.com
Marvel Snap Locations List - Game Rant
Opens in a new window

thegamer.com
Marvel Snap: Complete Locations Guide - TheGamer
Opens in a new window

medium.com
The Animated Monte-Carlo Tree Search (MCTS) | by Thomas Kurbiel - Medium
Opens in a new window

researchgate.net
A Taxonomy of Collectible Card Games from a Game-Playing AI Perspective
Opens in a new window

eclecticdevlog.com
Snapdragon, Introduction - Eclectic Dev Log
Opens in a new window

dualshockers.com
Marvel Snap: 10 Ways To Predict Your Opponent's Moves - DualShockers
Opens in a new window

keithburgun.net
Marvel Snap is a testament to the power of ruleset design – KEITH BURGUN GAMES
Opens in a new window

pcgamesn.com
Best Marvel Snap decks 2025 - PCGamesN
Opens in a new window

blakeir.com
Here's why Marvel Snap's game design is so ingenious
Opens in a new window

reddit.com
Strategy Talk: Game theory only decision, They can't play, you can't win, they don't know that

A Practical Roadmap for Your Marvel Snap AI

The key to success here is not to try and build the most complex AI from the outset, but to start with a solid foundation and add layers of intelligence incrementally.

Phase 1: Building the Game Engine (The Sandbox)

Before any AI can play, it needs a game to play in. The first and most critical step is to create a digital copy, or a "simulator," of *Marvel Snap*. This engine doesn't need fancy graphics; it just needs to understand and enforce the rules of the game.

- **What the Engine Must Do:**
  - **Represent the Game State:** It needs to track everything: the cards in each player's hand and deck, the three locations and their effects, each player's available energy per turn, and the power on the board.
  - **Enforce Rules:** The engine must correctly implement card costs, abilities (On Reveal, Ongoing, etc.), location effects, and the turn structure (six turns, simultaneous play).
  - **Manage Actions:** It must know what moves are legal at any given moment (e.g., which cards can be played with the current energy) and be able to execute those moves, updating the game state accordingly.
  - **Determine a Winner:** At the end of turn six, it must be able to calculate the power at each location and declare a winner based on the "win two of three" rule.
- **Coding Environment Recommendation:**
  - **Python:** I strongly recommend using Python for this project. It is renowned for being beginner-friendly and is the language of choice for the vast majority of AI and machine learning development. You can build a functional command-line version of the game without getting bogged down in complex graphical interfaces.
  - **Leverage Open Source:** To save significant time, you can look at open-source *Marvel Snap* simulators available on platforms like GitHub. Projects like `Marvel-Snap-Simulator` written in Python can provide an excellent starting point or a valuable reference for how to structure your own game engine.

Phase 2: The Simple Heuristic AI (Your First Opponent)

Once your engine is working, you can create your first, simple AI. This AI will make decisions based on a "heuristic," which is just a fancy term for a rule of thumb that evaluates how good a particular game state is.

- **The Heuristic Function:** Create a simple function that scores the board. A basic but effective starting point could be: `Score = (Your Total Power on Board) - (Opponent's Total Power on Board)`.
- **How it Plays:** On its turn, the AI will look at every possible legal move it can make. For each move, it will simulate the outcome and use your heuristic function to score the resulting board state. It will then choose the move that leads to the highest score. This approach is straightforward to implement and will give you a baseline AI that can play the game, even if it's not a master strategist.

Phase 3: The Monte Carlo AI (A Smarter Player)

This is where your AI starts to get genuinely clever, without requiring a supercomputer. The **Monte Carlo search** method is exceptionally well-suited for *Marvel Snap* given your constraints.

- **The Core Idea:** Instead of relying on a potentially flawed heuristic, this method uses the power of random simulation. For every possible legal move the AI can make from the current state, it plays out the rest of the game randomly hundreds or even thousands of times. The move that leads to the highest number of random victories is considered the best move.
- **Why It Works for *Snap* and Your Surface Pro:**
  - The game is short (6 turns) and the number of possible plays each turn is limited, which keeps the total number of simulations manageable.
  - This method is primarily CPU-based and doesn't require a powerful graphics card, making it ideal for your hardware. You can adjust the number of simulations (e.g., start with 50 per move) to balance performance and decision quality.
- **Handling Hidden Information:** A key challenge in card games is not knowing the opponent's hand. MCTS handles this with a technique called **determinization**. In simple terms, for each random simulation, the AI "pretends" it knows the opponent's hand and deck by dealing them random, plausible cards from the pool of all cards that haven't been seen yet. By running thousands of simulations with different random "guesses" for the opponent's hand, the AI gets a robust statistical average of how good each of its moves is.

Implementing Self-Play and Learning

You correctly identified that the AI needs to play itself to learn. This is a cornerstone of modern game AI.

1. **Testing and Tuning:** The most immediate use of self-play is to have your Monte Carlo AI play against itself or against your simpler heuristic AI. This is an excellent way to test its effectiveness and see if your changes are making it smarter.
2. **Evolving Decks (The Next Step):** Once your AI can play a given deck competently, you can use self-play to discover the *best decks*. This is a concept known as a genetic algorithm.
   - Generate a "population" of random 12-card decks.
   - Have the decks play against each other in a tournament, with your AI piloting them.
   - The decks with the highest win rates are considered the "fittest."
   - "Reproduce" by creating new decks that are combinations of the fittest decks, with a small chance of "mutation" (swapping in a random new card).
   - Over many generations, your system will evolve highly optimized decks for the base game meta.

For your goal of building a game-playing bot on a personal computer in a limited time, this is like using a sledgehammer to crack a nut. The Monte Carlo approach is far more practical, achievable, and will give you excellent results and a fantastic learning experience within your constraints.

By following this phased plan, you can start with a simple, working game, build a basic AI in a matter of hours, and then progress to a genuinely intelligent agent that can discover its own winning strategies through self-play.

Building a game AI for a more complex card game. Any guidance on which direction is most feasible? - Reddit
reddit.com/r/gameai/comments/g85fnt/building_a_game_ai_for_a_more_complex_card_game
Opens in a new window

How would one approach developing an AI for a trading card game [closed]
gamedev.stackexchange.com/questions/33972/how-would-one-approach-developing-an-ai-for-a-trading-card-game
Opens in a new window

Snapdragon, Introduction - Eclectic Dev Log
eclecticdevlog.com/posts/snapdragon_intro
Opens in a new window

Building a game AI for a more complex card game. Any guidance on which direction is most feasible? - Reddit
reddit.com/r/gameai/comments/g85fnt/building_a_game_ai_for_a_more_complex_card_game
Opens in a new window

How would one approach developing an AI for a trading card game [closed]
gamedev.stackexchange.com/questions/33972/how-would-one-approach-developing-an-ai-for-a-trading-card-game
Opens in a new window

Marvel Snap in Python: CLI Edition! Build Your First Game Today! - YouTube
youtube.com/watch?v=aKUXXigS0xM
Opens in a new window

CREATE YOUR OWN Marvel Snap GAME WITH Python TODAY! - YouTube
youtube.com/watch?v=0o8dM92q_-8
Opens in a new window

georgelid06/Marvel-Snap-Simulator - GitHub
github.com/georgelid06/Marvel-Snap-Simulator
Opens in a new window

prfalken/Marvel-snap-simulator - GitHub
github.com/prfalken/Marvel-snap-simulator
Opens in a new window

A Three Slots Resource Allocation Study Definitely Not In Marvel Snap | leriohub
leriohub.com/a-three-slots-resource-allocation-study-definitely-not-in-marvel-snap
Opens in a new window

How would one approach developing an AI for a trading card game [closed]
gamedev.stackexchange.com/questions/33972/how-would-one-approach-developing-an-ai-for-a-trading-card-game
Opens in a new window

Snapdragon, Introduction - Eclectic Dev Log
eclecticdevlog.com/posts/snapdragon_intro
Opens in a new window

ML | Monte Carlo Tree Search (MCTS) - GeeksforGeeks
geeksforgeeks.org/machine-learning/ml-monte-carlo-tree-search-mcts
Opens in a new window

The Animated Monte-Carlo Tree Search (MCTS) | by Thomas Kurbiel - Medium
medium.com/data-science/the-animated-monte-carlo-tree-search-mcts-c05bb48b018c
Opens in a new window

Snapdragon, Introduction - Eclectic Dev Log
eclecticdevlog.com/posts/snapdragon_intro
Opens in a new window

Snapdragon, Introduction - Eclectic Dev Log
eclecticdevlog.com/posts/snapdragon_intro
Opens in a new window

Building a game AI for a more complex card game. Any guidance on which direction is most feasible? - Reddit
reddit.com/r/gameai/comments/g85fnt/building_a_game_ai_for_a_more_complex_card_game
Opens in a new window

Snapdragon, The Basic Assumption(s) - Eclectic Dev Log
eclecticdevlog.com/posts/snapdragon_basic_assumption
Opens in a new window

Monte-carlo search tree with hidden information - Stack Overflow
stackoverflow.com/questions/75715896/monte-carlo-search-tree-with-hidden-information
Opens in a new window

[Self-Play Reinforcement Learning for Strategic Games | by Amit Yadav | Biased-Algorithms](medium.com/biased-algorithms/self-play-reinforcement-le)
medium.com/biased-algorithms/self-play-reinforcement-le