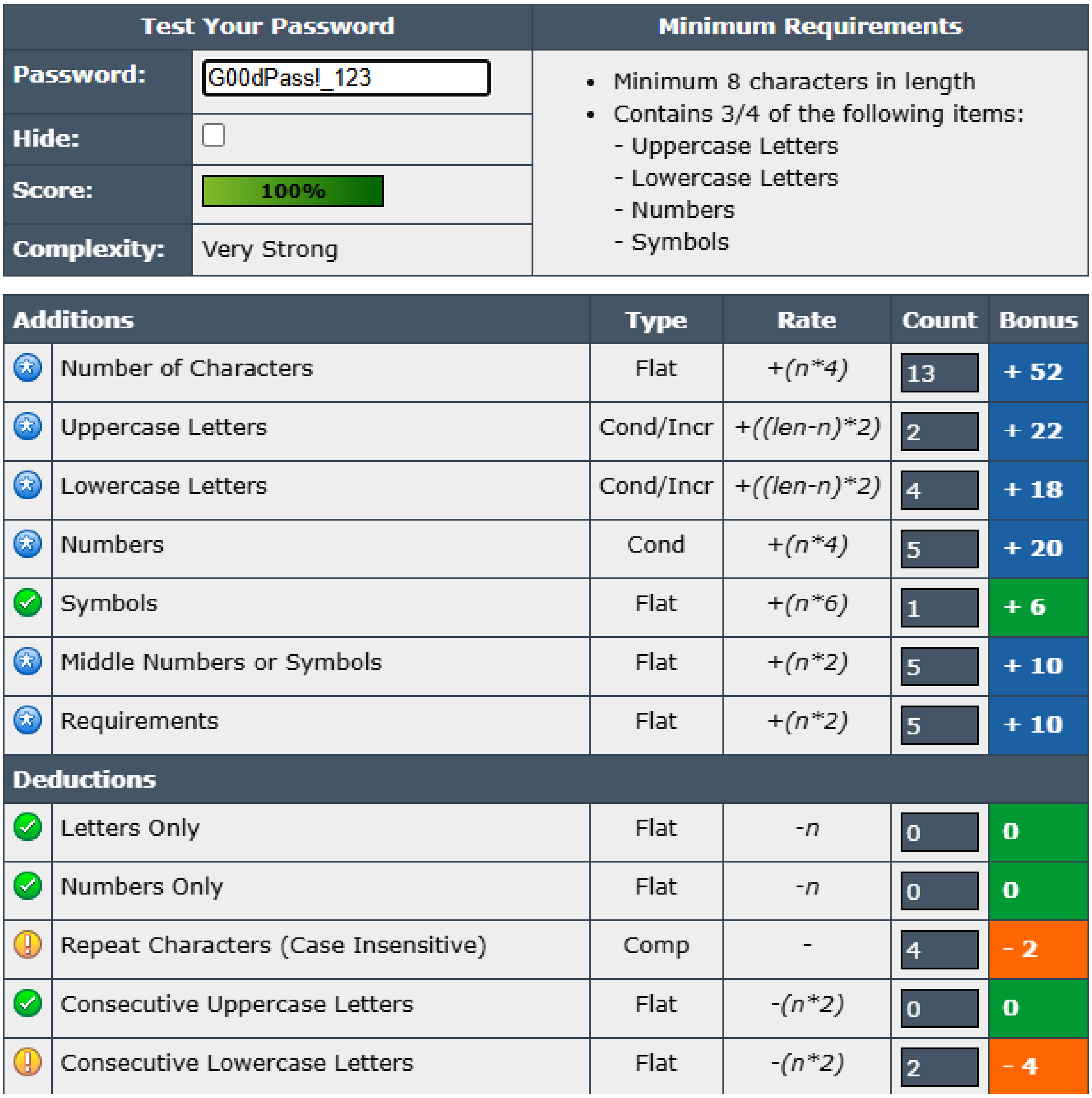
TASK 6

Objective: Understand what makes a password strong and test it against password strength tools. Tools: Online free password strength checkers (e.g., passwordmeter.com)

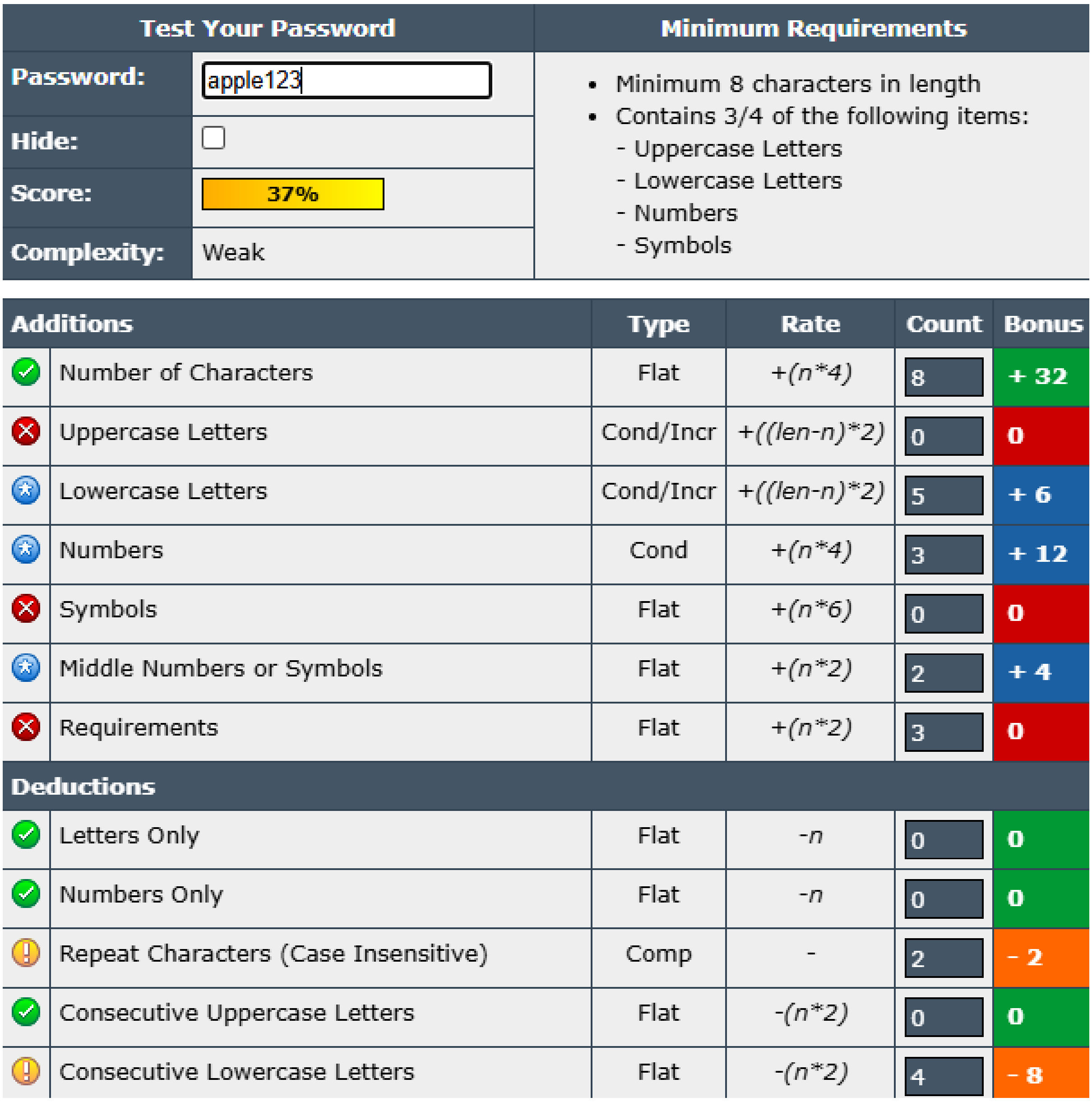


Test Your Password	
Password:	G00dPass!_123
Hide:	□
Score:	100%
Complexity:	Very Strong

Minimum Requirements

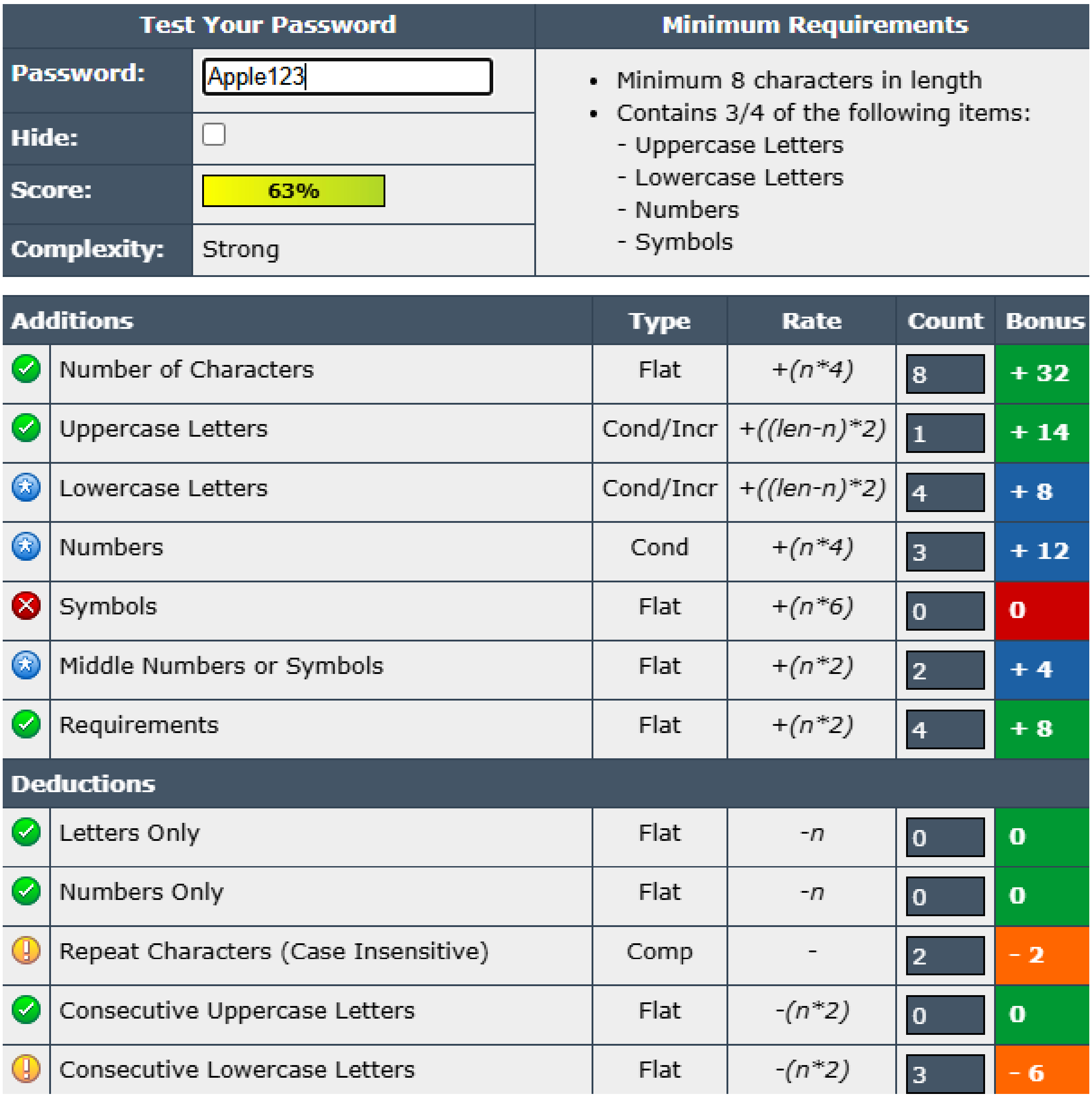
- Minimum 8 characters in length
- Contains 3/4 of the following items:
  - Uppercase Letters
  - Lowercase Letters
  - Numbers
  - Symbols

Additions		Type	Rate	Count	Bonus
★	Number of Characters	Flat	+(n*4)	13	+ 52
★	Uppercase Letters	Cond/Incr	+((len-n)*2)	2	+ 22
★	Lowercase Letters	Cond/Incr	+((len-n)*2)	4	+ 18
★	Numbers	Cond	+(n*4)	5	+ 20
✓	Symbols	Flat	+(n*6)	1	+ 6
★	Middle Numbers or Symbols	Flat	+(n*2)	5	+ 10
★	Requirements	Flat	+(n*2)	5	+ 10
Deductions					
✓	Letters Only	Flat	-n	0	0
✓	Numbers Only	Flat	-n	0	0
!	Repeat Characters (Case Insensitive)	Comp	-	4	- 2
✓	Consecutive Uppercase Letters	Flat	-(n*2)	0	0
!	Consecutive Lowercase Letters	Flat	-(n*2)	2	- 4



Test Your Password		Minimum Requirements
Password:	apple123	- Minimum 8 characters in length - Contains 3/4 of the following items: - Uppercase Letters - Lowercase Letters - Numbers - Symbols
Hide:	□	
Score:	37%	
Complexity:	Weak	

	Additions	**Type**	**Rate**	**Count**	**Bonus**
✓	Number of Characters	Flat	+(n*4)	8	+ 32
✗	Uppercase Letters	Cond/Incr	+((len-n)*2)	0	0
★	Lowercase Letters	Cond/Incr	+((len-n)*2)	5	+ 6
★	Numbers	Cond	+(n*4)	3	+ 12
✗	Symbols	Flat	+(n*6)	0	0
★	Middle Numbers or Symbols	Flat	+(n*2)	2	+ 4
✗	Requirements	Flat	+(n*2)	3	0
	Deductions				
✓	Letters Only	Flat	-n	0	0
✓	Numbers Only	Flat	-n	0	0
!	Repeat Characters (Case Insensitive)	Comp	-	2	- 2
✓	Consecutive Uppercase Letters	Flat	-(n*2)	0	0
!	Consecutive Lowercase Letters	Flat	-(n*2)	4	- 8



Test Your Password	
Password:	Apple123
Hide:	□
Score:	63%
Complexity:	Strong

Minimum Requirements

- Minimum 8 characters in length
- Contains 3/4 of the following items:
  - Uppercase Letters
  - Lowercase Letters
  - Numbers
  - Symbols

	Additions	Type	Rate	Count	Bonus
✓	Number of Characters	Flat	+(n*4)	8	+ 32
✓	Uppercase Letters	Cond/Incr	+((len-n)*2)	1	+ 14
★	Lowercase Letters	Cond/Incr	+((len-n)*2)	4	+ 8
★	Numbers	Cond	+(n*4)	3	+ 12
✗	Symbols	Flat	+(n*6)	0	0
★	Middle Numbers or Symbols	Flat	+(n*2)	2	+ 4
✓	Requirements	Flat	+(n*2)	4	+ 8
	Deductions				
✓	Letters Only	Flat	-n	0	0
✓	Numbers Only	Flat	-n	0	0
!	Repeat Characters (Case Insensitive)	Comp	-	2	- 2
✓	Consecutive Uppercase Letters	Flat	-(n*2)	0	0
!	Consecutive Lowercase Letters	Flat	-(n*2)	3	- 6

Identify best practices for strong passwords

- Use **12+ characters** when possible.
- Combine **uppercase, lowercase, numbers, and special characters**.
- Avoid dictionary words or predictable patterns.
- Use **passphrases** (random but memorable sentence-like combinations).
- Avoid reusing passwords across sites.
- Use a **password manager** for storage.

Tips learned from evaluation

- Adding just **one special character** significantly increases crack time.
- Increasing length from **8 to 12 characters** boosts security exponentially.
- Randomness > complexity rules (e.g., `CorrectHorseBatteryStaple` is stronger than `P@ssw0rd!`).
- Avoid personal information like names, birthdays.

Common password attacks

- **Brute Force** – Tries every possible combination; long & random passwords are the best defense.
- **Dictionary Attack** – Uses common words or leaked password lists; avoid real words.
- **Credential Stuffing** – Uses stolen passwords on multiple sites; never reuse passwords.
- **Phishing** – Tricks you into revealing the password; be cautious with suspicious links.

How password complexity affects security

- **Length** is the most important factor — even a simple but long password takes exponentially longer to crack.
- **Character variety** (upper, lower, numbers, symbols) expands the possible combinations, slowing brute force attacks.
- **Randomness** prevents guessing or dictionary-based attacks.
- **Unique passwords** reduce the damage from breaches.