**Assignment**

**CSA0814– Python Programming**

|  |  |
| --- | --- |
| **Register Number** | **192311291** |
| **Name** | **Sayed Fazal** |

**Title: File Integrity Checker**

**Problem Statement:** Write a Python program that calculates checksums (e.g., MD5, SHA-256) for files and verifies their integrity by comparing them with stored values, detecting changes or corruption

**Code:**

**import os**

**import hashlib**

**def calculate\_checksum(file\_path):**

**"""**

**Calculate the SHA-256 checksum of a file**

**"""**

**with open(file\_path, 'rb') as f:**

**file\_content = f.read()**

**checksum = hashlib.sha256(file\_content).hexdigest()**

**return checksum**

**def check\_file\_integrity(file\_path, expected\_checksum):**

**"""**

**Check if the file's checksum matches the expected checksum**

**"""**

**actual\_checksum = calculate\_checksum(file\_path)**

**if actual\_checksum == expected\_checksum:**

**print(f"File {file\_path} is intact. Checksum matches.")**

**else:**

**print(f"File {file\_path} has been modified. Checksum mismatch.")**

**def main():**

**# Example usage:**

**file\_path = "/content/example.txt "**

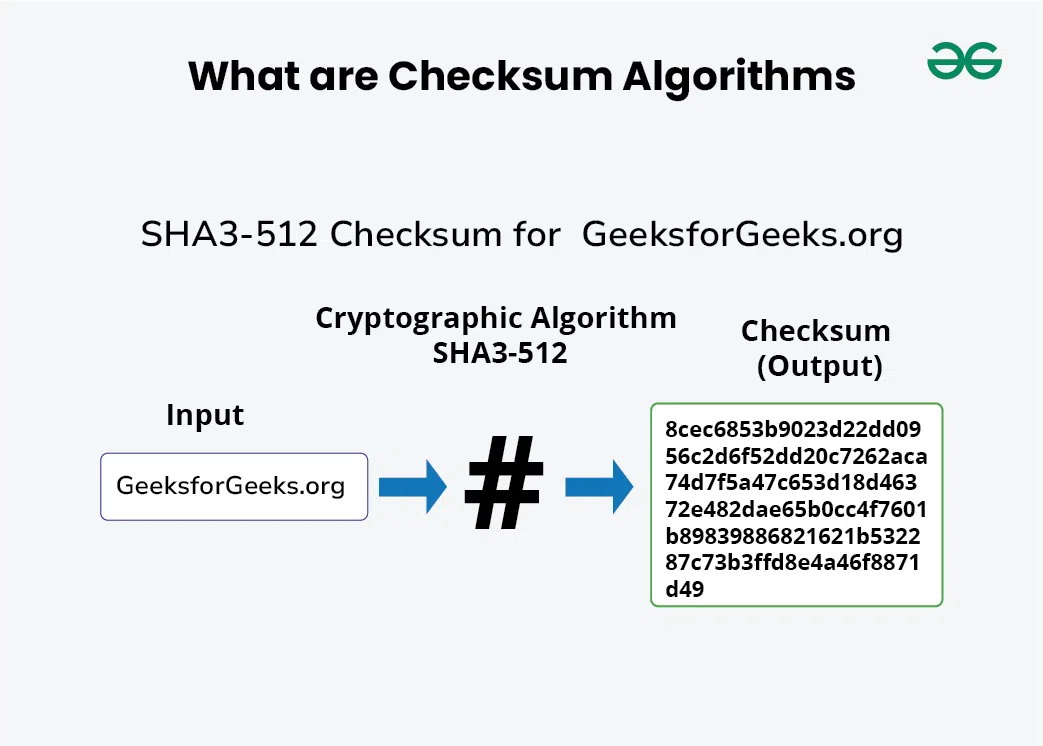
**expected\_checksum = "8cec6853b9023d22dd0956c2d6f52dd20c7262aca74d7f5a47c653d18d46372e482dae65b0cc4f7601689839886821621b532287c73b3ffd8e4a46f8871d49"**

**check\_file\_integrity(file\_path, expected\_checksum)**

**if \_\_name\_\_ == "\_\_main\_\_":**

**main()**

**Output Screen Shots:**



**Conclusion:**

This Python program demonstrates a basic file integrity checker that calculates checksums for a file using the MD5 and SHA-256 algorithms and verifies their integrity by comparing them with stored values.