**Protokoll zum Programm myFind**

Das Programm myFind ist ein benutzerfreundliches Tool, das durch den Befehl

./myfind [-R] [-i] searchpath filename1 [filename2] …[filenameN] aufgerufen wird. Es ermöglicht die parallele Suche nach verschiedenen Dateien in einem bestimmten Ordner. In diesem Protokoll werden die Hauptkonzepte der Parallelisierung und die notwendige Synchronisation beschrieben.

**Parallelisierung durch Kindprozesse**

Für jede zu suchende Datei wird ein separater Kindprozess erstellt. Dies ermöglicht die parallele Suche nach mehreren Dateien. Dies wird durch den fork()-Befehl erreicht, der einen neuen Prozess erstellt, der eine Kopie des aktuellen Prozesses ist.

**Argumentenverarbeitung mit getopt()**

Das Programm verwendet die getopt()-Funktion, um Befehlszeilenargumente zu analysieren und Optionen wie -R und -i zu erkennen. Dies erleichtert die Handhabung von Argumenten und stellt sicher, dass sie in beliebiger Reihenfolge angegeben werden können.

**Synchronisation zwischen Eltern- und Kindprozessen**

Der Elternprozess wartet, bis alle Kindprozesse ihre Suche abgeschlossen haben, bevor er selbst beendet wird. Dies wird durch den wait(NULL)-Befehl erreicht. Dies stellt sicher, dass keine "Zombie"-Prozesse zurückbleiben.

**Rekursive Suche**

Wenn die Option **-R** aktiviert ist, durchsucht das Programm rekursiv alle Unterverzeichnisse des angegebenen Pfades. Dies wird durch einen rekursiven Aufruf der searchFile Funktion implementiert.

**Groß- und Kleinschreibung-unabhängige Suche**

Mit der Option **-i** wird die Suche unabhängig von Groß- und Kleinschreibung durchgeführt. Dies wird durch die caseInsensitiveMatch Funktion erreicht.

**Verzeichniszugriff und Dateisuche**

Das Programm nutzt die Funktionen opendir und readdir, um auf Verzeichnisse zuzugreifen und deren Einträge zu durchlaufen. Bei Übereinstimmungen wird der Pfad, die Prozess-ID und der Dateiname ausgegeben.

**Fehlerbehandlung**

Das Programm enthält Fehlerbehandlungsroutinen, um ordnungsgemäß zu funktionieren und dem Benutzer nützliche Meldungen zu liefern, z.B. bei nicht öffnenden Verzeichnissen oder fehlgeschlagenen Kindprozessen.