

## 1 Einleitung

Herzlich willkommen zu dieser Studie zur Analyse des Einflusses von Entwicklern auf den Energieverbrauch von Programmen. Im Folgenden werden Sie verschiedene Aufgaben bearbeiten und die von uns bereitgestellten Werkzeuge nutzen. Ihr Ziel ist es, den Energieverbrauch der vorgegebenen Programme zu optimieren. Sollten während der Bearbeitung Fragen oder Unklarheiten auftreten, zögern Sie bitte nicht, diese direkt an uns zu richten.

## 2 Arbeitsumgebung

Vor sich finden Sie einen Rechner, den wir extra für die Studie eingerichtet haben. Zusätzlich zu diesem Aufgabenblatt haben Sie einen Zettel mit Zugangsdaten zu ihrem Account erhalten. Loggen Sie sich bitte damit ein.

Nach dem Login öffnet sich automatisch der Dateimanager in dem Ordner `/Dokumente/study` Innerhalb

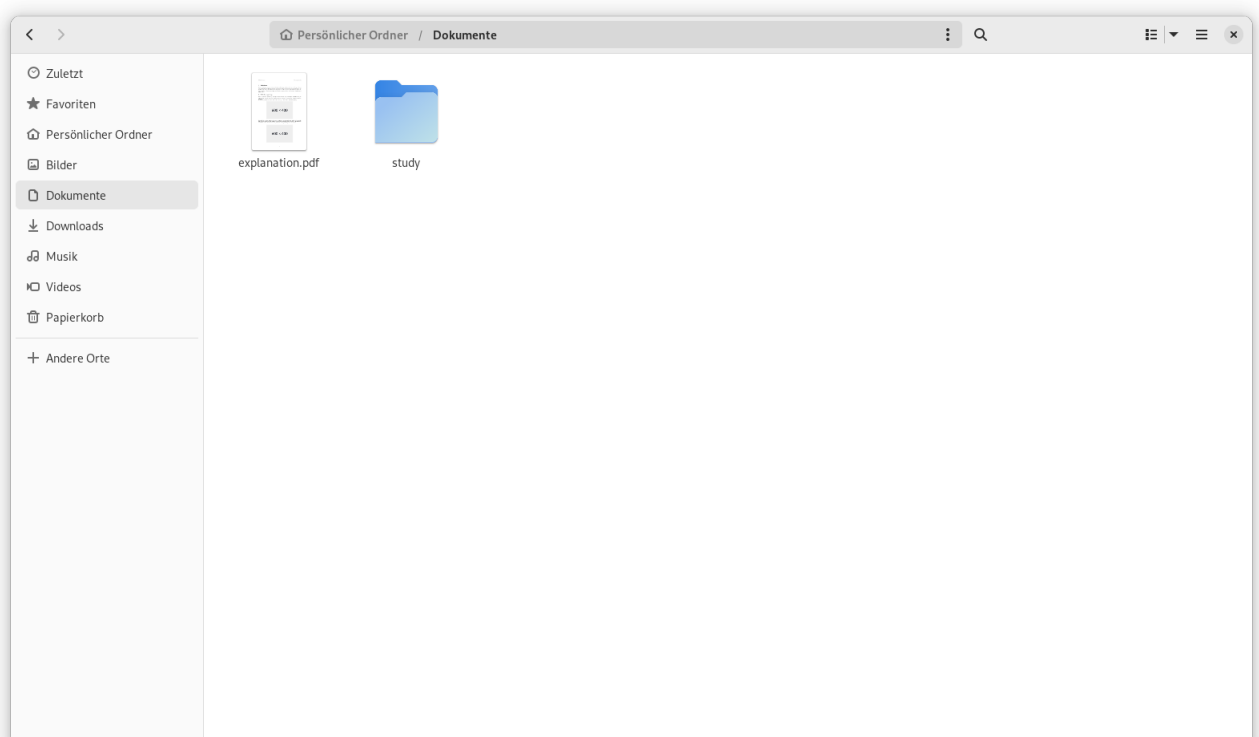


Abbildung 1: Der Ordner study enthält Ihre Arbeitsumgebung

dieses Ordners finden den Unterordner `tasks`, der die einzelnen Projekte enthält die Sie im Rahmen der Studie bearbeiten sollen. Zusätzlich dazu befinden sich in dem Ordner neben einer digitalen Kopie dieses Blatts weitere hilfreiche Dokumente, um die Bearbeitung der Aufgaben zu unterstützen.

In der Schnellzugriffsleiste haben wir bereits Verknüpfungen zu der verwendbaren Software installiert unter anderem auch Visual Studio Code um die Aufgaben bearbeiten zu können. Wir haben in Visual Studio Code einige Extensions installiert, um die Bearbeitung der Aufgaben sinnvoll zu ermöglichen.

**Bitte nutzen Sie für die Bearbeitung der Aufgaben ausschliesslich Visual Studio Code!**

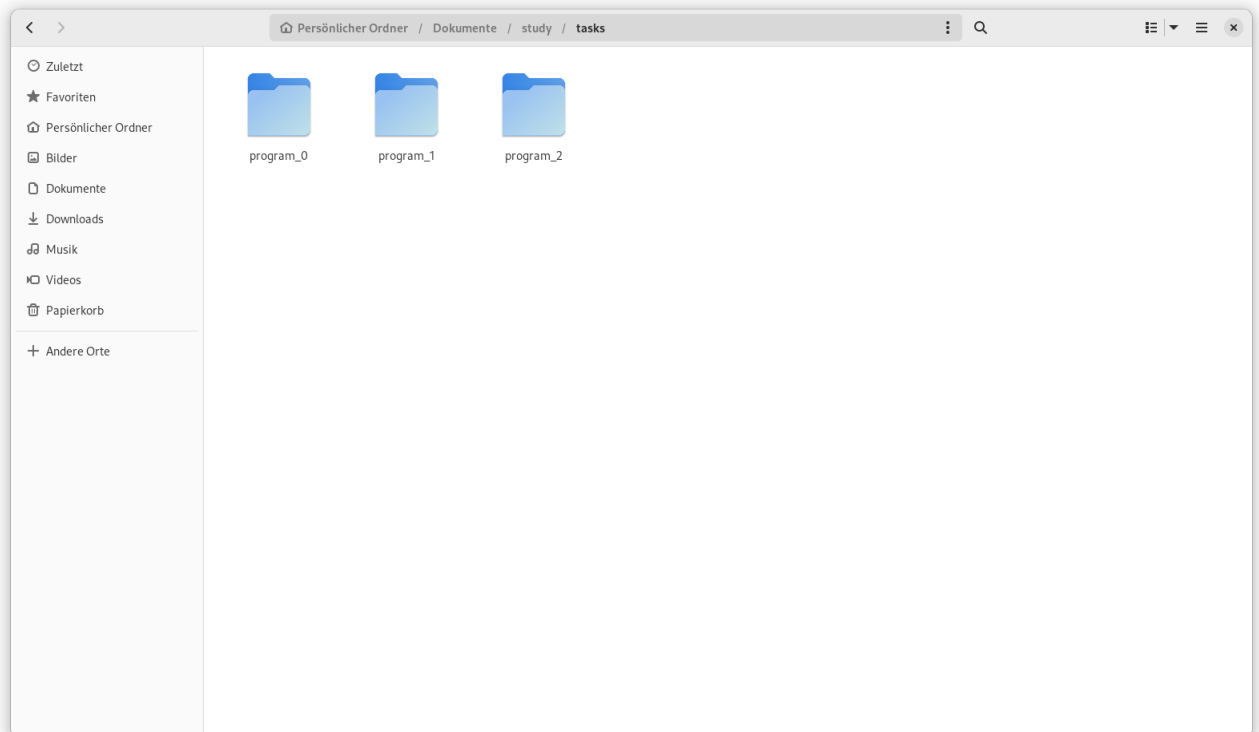


Abbildung 2: Der Ordner Tasks mit den einzelnen Programmen

### 3 Buildscript

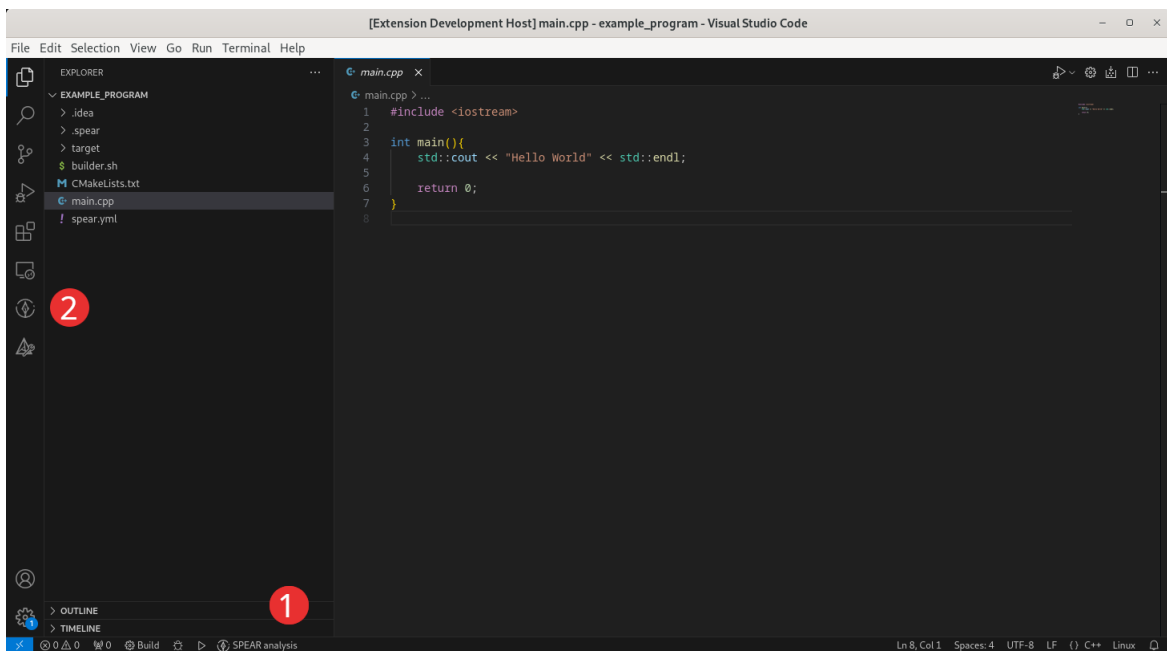
Um die Bearbeitung der Aufgaben so einfach wie möglich zu gestalten, stellen wir Ihnen ein Shell-Script bereit, das die Kompilierung der Aufgaben übernimmt. In jedem Aufgaben-Unterverzeichnis finden Sie die Datei `builder.sh`. Diese erlaubt folgende Befehle:

- `./builder.sh build`  
Kompiliert die Anwendung und erzeugt eine ausführbare Anwendung im Ordner `target/`
- `./builder.sh clean`  
Löscht das `target/` Verzeichnis, um einen Clean-Build, ohne gecachte Dateien zu ermöglichen

### 4 SPEAR-Viewer

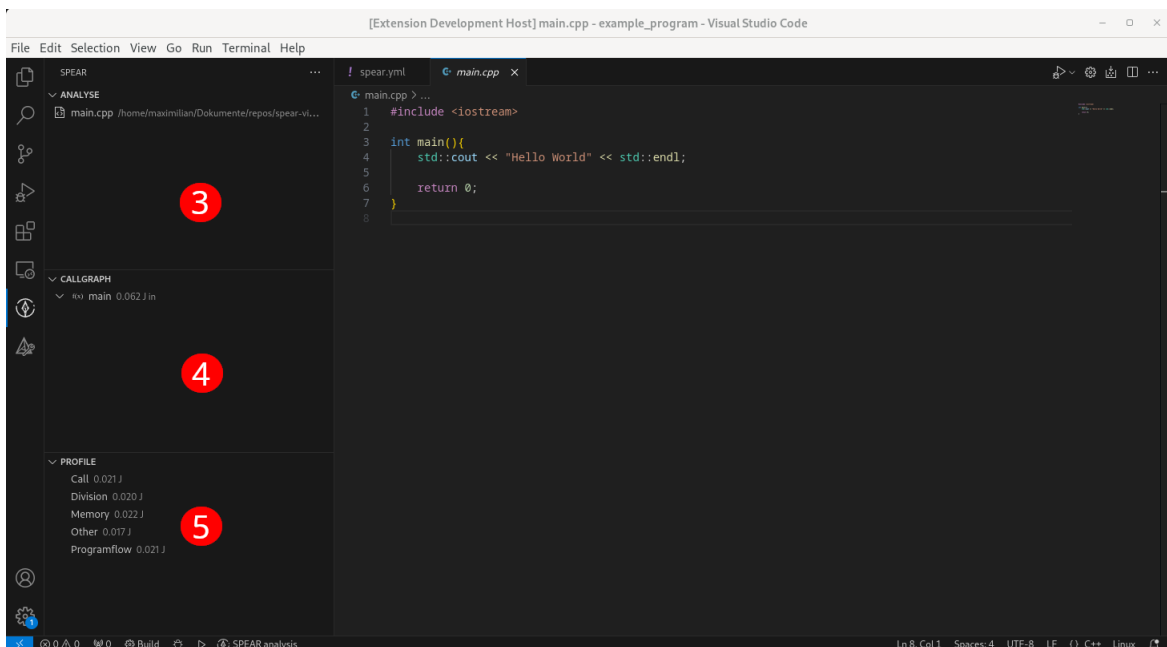
Um die Bearbeitung der Aufgaben zu unterstützen, stellen wir Ihnen den **SPEAR-Viewer** zur Verfügung. Dabei handelt es sich um eine Extension für Visual Studio Code, die es Ihnen ermöglicht, Details zum Energieverbrauch eines Programms zu erhalten. Im Folgenden werden die wichtigsten Komponenten der Extension genauer erläutert:

Sobald Sie Visual Studio Code starten, sehen sie in der linken Sidebar, sowie in der FuSSleiste jeweils ein neues Icon:

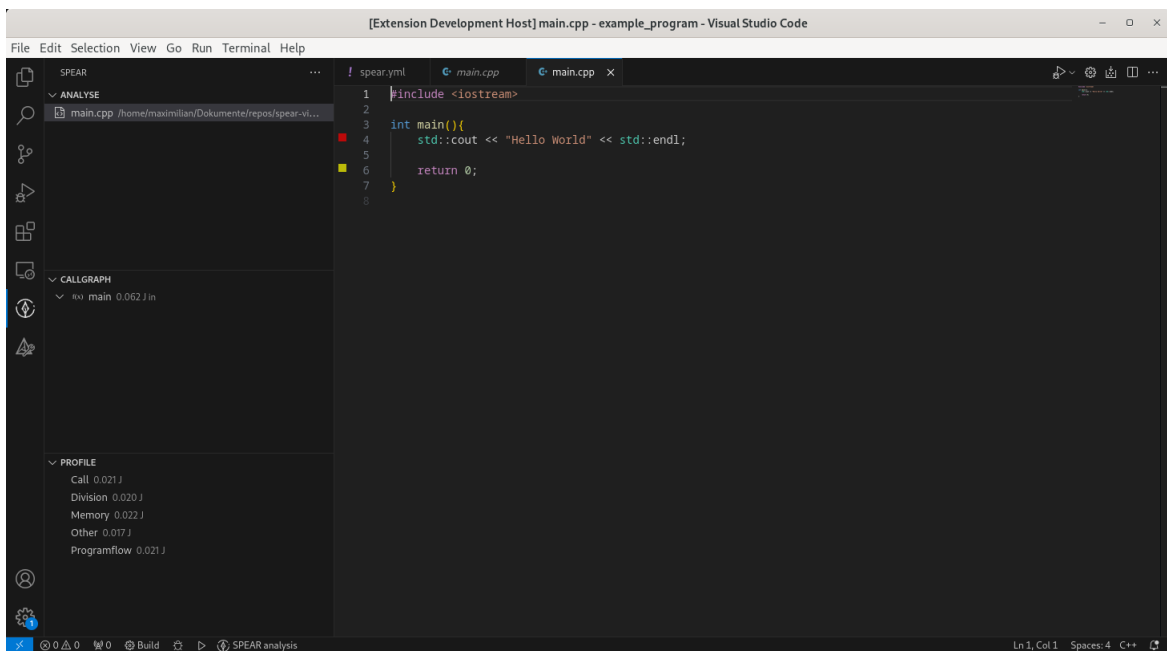


In (1) können Sie eine neue Energieanalyse Ihres Programms starten. Die Extension berechnet dann statisch den Energieverbrauch ihres Programms und leitet Sie auf die detaillierte Ansicht (2) weiter. Nach jeder Änderung an Ihrem Programm müssen Sie die Analyse erneut ausführen, um diese auf den aktuellen Stand zu bringen.

In (2) sehen Sie Details zu dem Energieverbrauch ihrer Anwendung:



In (3) werden alle Dateien aufgelistet, die in die Energieberechnung einbezogen werden. Neben dem Namen der Datei wird Ihnen auch der Pfad angegeben. Durch doppeltes Klicken auf den Dateinamen öffnet sich die Energie-Ansicht zu der jeweiligen Datei. In dieser Ansicht wird der Energieverbrauch der einzelnen Code-Elemente farblich visualisiert:

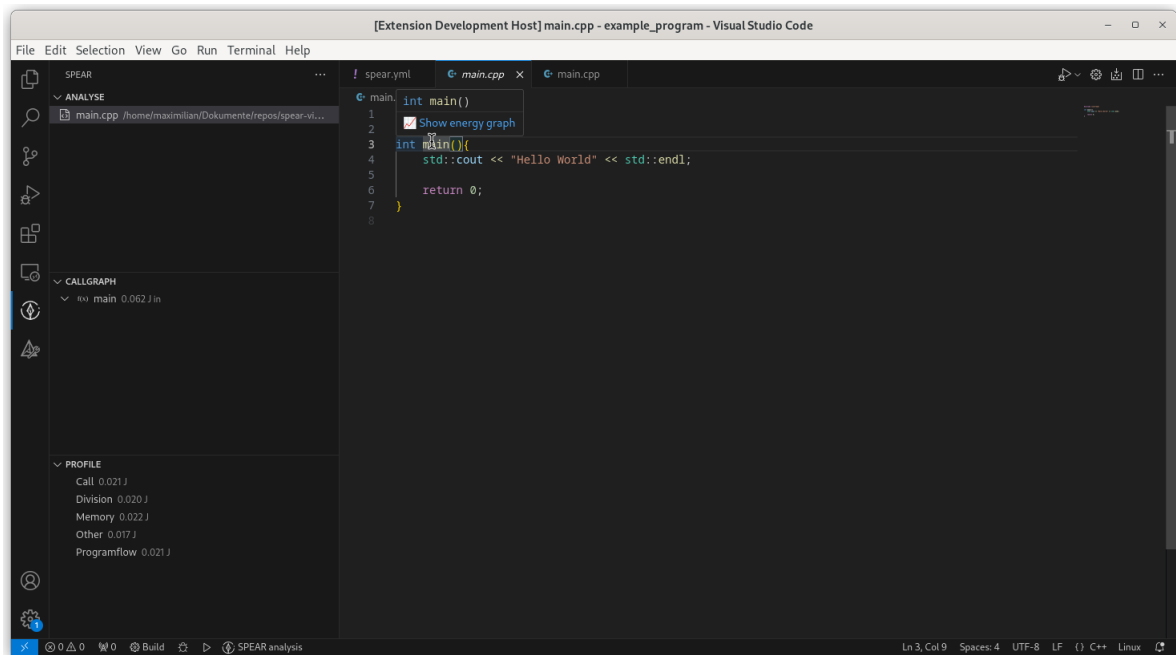


Durch das Hovern über eine Zeile wird der berechnete Energieverbrauch angezeigt,

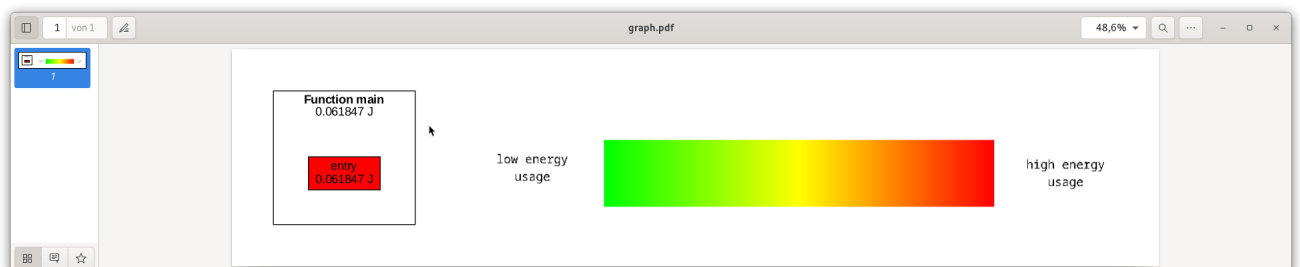
In (4) erhalten Sie eine Darstellung des Callgraph ihres Programms. Die einzelnen Funktionen werden dabei mit dem berechneten Energieverbrauch annotiert. Die Knoten des Graphen können hierarchisch durch einen Klick ein bzw. ausgeklappt werden.

Abschnitt (5) bietet Ihnen einen groben Einblick des Energieverbrauchs der einzelnen Instruktionen. Die Instruktionen Ihres Programms sind dabei in Kategorien gruppiert. Jede Gruppe hat einen charakteristischen Energiewert, der spezifisch für das vorliegende System ist.

Neben den einzelnen Metriken und der farblichen Visualisierung des Codes bietet der SPEAR-Viewer auch die Möglichkeit, einen Callgraph anzuzeigen, der farblich den Energieverbrauch der einzelnen Funktionen darstellt. Öffnen Sie dazu die Sourcecode-Datei, die sie analysieren wollen, und hovern über den Namen der Funktion, dessen Callgraph erzeugt werden soll:



Durch einen Klick auf den Text “Show energy graph” öffnet sich automatisch ein PDF, das den Callgraph darstellt. Zusätzlich dazu erhalten Sie eine Farbskala, die den Energieverbrauch beschreibt.



## 5 Aufgabe

Ihre Aufgabe ist nun die Bearbeitung der einzelnen Programme in dem Ordner **tasks**. Ziel ist dabei die Reduktion des Energieverbrauchs der Programme. Sie dürfen diese nach Belieben abändern, ohne die Grundfunktion zu beeinflussen. D.h. ein **Hello-World**-Programm sollte nach ihrer Bearbeitung weiterhin “Hello World” ausgeben. Bearbeiten Sie die Programme nacheinander und achten Sie auf die Compilierbarkeit, die sie über das Script **builder.sh** testen können. Sollten Sie den Energieverbrauch nicht weiter optimieren können, wechseln sie gerne zum nächsten Programm. Zum Öffnen eines Programms nutzen sie am Besten die eingebaute Funktion von Visual Studio Code über **File -> Open Folder**. Zur Bearbeitung der drei Programme haben Sie 30 Min. Zeit. Sollten Sie früher fertig sein, melden Sie sich bitte.