

1 Heap vs. Stack

Schreiben Sie eine Funktion, die folgende Informationen über einen Integer-Zeiger ausgibt:

- Den Wert
- Zeiger-Adresse

Jeweils eine Variable auf dem Heap und auf dem Stack soll erzeugt und initialisiert werden. Anschließend, sollen der Wert und die Speicheradresse mit der geschriebener Funktion ausgegeben werden. Denken Sie bitte an die Speicherfreigabe und Fehlerbehandlung, wenn der Speicher dynamisch reserviert wird.

2 Dynamische Speicherallokation

Implementieren Sie eine Funktion, die ein Integer Array um 1 Element erweitert und mit 0 initialisiert. Schreiben sie auch eine Funktion, die den Inhalt ausgibt. Denken Sie bitte an die Speicherfreigabe und Fehlerbehandlung, wenn der Speicher dynamisch reserviert wird.

Hinweis: `malloc()` oder die `calloc()` Funktion können dazu benutzt werden, um Speicher für ein Array auf dem Heap zu reservieren.

3 Memory Leaks (Bewertete Aufgabe)

Erzeugen Sie eigene Beispiele für die folgenden Fehlersituationen:

- Speicherleck
- Variablennutzung nach `free()`
- Freigabe vom nicht dynamisch reservierten Speicher
- Zugriff auf nicht reservierten Speicher
- Mehrfacher Aufruf von `free()`

Begründen Sie in einem kurzen Kommentar warum es sich um ein Fehler handelt.

Abgabe

Abzugeben ist ein gemäß den bekannten Richtlinien erstelltes und benanntes Archiv. Das enthaltene und gewohnt benannte Verzeichnis soll folgenden Inhalt haben:

- Alle Quellen, aus denen Ihr Programm besteht (`exercise3.c`); gut dokumentiert (Kommentare im Code!)

Senden Sie Ihre Abgabe an `cp-abgabe@wr.informatik.uni-hamburg.de`.