

Algorithmen und Programmierung

Potenzfunktion und Verwandte (Trainingsaufgabe)

Schreiben Sie eine Datei `power.c`, die folgende Funktionen **ohne** die Verwendung von Schleifen (Schlüsselwörter `for`, `while`, `do`), d.h. wenn nötig **rekursiv**, implementiert.

1. n -te Potenz der ganzen Zahl x :

$$x^n$$

```
int power(int x, unsigned int n)
```

2. Funktionswert des Polynoms:

$$\sum_{i=0}^n x^i = 1 + x + x^2 \dots + x^n$$

```
int poly1(int x, unsigned int n)
```

3. Funktionswert des Polynoms:

$$\sum_{i=0}^n (-1)^i \cdot x^i = 1 - x + x^2 \dots \pm x^n$$

```
int polyAlter(int x, unsigned int n)
```

Achten Sie darauf, dass die Datei zur Abgabe *keine* `main`-Funktion beinhalten darf.

Weiterführende Überlegungen:

- Wieviele Rechenoperationen in Abhängigkeit von n benötigt Ihre Berechnung der Potenzfunktion?
- Gibt es schnellere Algorithmen?
- Wieviele Rechenoperationen in Abhängigkeit von n benötigen Sie zur Berechnung des Polynoms?
- Welche Einsparung ist mit Hilfe des *Horner*-Schemas möglich?
- Kann die Berechnung für dieses spezielle Polynom noch weiter vereinfacht werden?

Hinweise zur Aufgabenstellung

Für die Lösung dieser Aufgabe benötigen Sie folgende Grundkenntnisse:

- Kontrollfluss in C
- Rekursion
- Funktionen in C
- Benutzung von `gcc`

Hinweise zur Abgabe

- Erstellen Sie eine ZIP- bzw. TGZ-Archivdatei, welche die geforderten Dateien enthält.
- Fügen Sie dem Archiv keine weiteren Dateien oder Ordner hinzu.
- Reichen Sie Ihre Lösung unter <https://osg.informatik.tu-chemnitz.de/submit> ein.
- Bis zum Abgabeende (Deadline), sofern gegeben, können beliebig neue Lösungen eingereicht werden, die die jeweils älteren Versionen ersetzen.
- Ihr Programm muss auf der Testmaschine übersetzbar sein. Deren Details sind auf dem OpenSubmit-Dashboard verfügbar.
- Ihre Lösung wird automatisch validiert. Sie werden über den Abschluss der Validierung per eMail informiert.