

## 0. Programmieren in C

Abgabe bis 30. April, 10:15 GMT+2.

### Aufgabe 0:

Die Fibonaccifolge ist definiert durch  $f_1 = f_2 = 1, f_{n+1} = f_n + f_{n-1}$ .  
Schreibe eine Funktion `unsigned long fibonacci(unsigned int n)` (in einer Datei `fibonacci.c`), die ein Glied der Folge berechnet!

### Aufgabe 1:

Das Verhältnis  $\Phi$  des goldenen Schnitts ist definiert durch  $\Phi = \frac{a}{b} = \frac{a+b}{a}$  für  $a > b > 0$ . Der Quotient aufeinanderfolgender Glieder der Fibonaccifolge konvergiert gegen das Verhältnis des goldenen Schnitts:  $\lim_{n \rightarrow \infty} \frac{f_{n+1}}{f_n} = \Phi$ .  
Schreibe eine Funktion `golden`, die das Verhältnis des goldenen Schnitts approximiert! Die Genauigkeit der Approximation soll vom einzigen Argument der Funktion abhängen, wobei höhere Werte tendenziell eine genauere Approximation ergeben sollen. Die Wahl der Datentypen für Rückgabewert und Argument sind zu begründen.

### Aufgabe 2:

Schreibe ein Programm, das eine Approximation des Verhältnisses des goldenen Schnitts ausgibt (in Form einer schönen und verständlichen Ausgabe)!  
Das Programm soll die Funktion aus der vorigen Aufgabe verwenden, und auf manchen heute üblichen Rechnern innerhalb weniger Sekunden terminieren.

### Aufgabe 3:

Schreibe ein Programm mit einem Unit Test für die Funktion aus Aufgabe 0!

Hinweise (gelten für dieses wie auch für alle weiteren Blätter):

- Der Code ist vollständig selbst zu schreiben.
- Der Code sollte dokumentiert sein. Wo nicht offensichtlich ist, wie der Code funktioniert, sollte er in Kommentaren erklärt sein.
- Die Abgabe erfolgt über Subversion. Hochzuladen ist der Code (.c, .cpp, etc), einschließlich Makefile, aber ohne generierte Dateien (.o, etc) in ein Unterverzeichnis `Blatt0` (entsprechend bei den folgenden Blättern dann `Blatt1`, etc).
- Relevant für die Korrektur ist die letzte, vor dem Abgabetermin in Subversion hochgeladene Version. Diese muss fehlerfrei auf Jenkins durchlaufen (andernfalls wird die Abgabe mit 0 Punkten bewertet).
- Bei Problemen, kann es hilfreich sein, im Forum zur Vorlesung zu fragen, falls dort noch keine einschlägige Frage gestellt wurde. Gute Fragen (also z.B. bei einer Fehlermeldung des Compilers diese, sowie den relevanten Code angeben statt einfach nur zu sagen, dass irgendetwas nicht funktioniert) haben bessere Chancen auf hilfreiche Antworten.
- Die Anforderungen an das Makefile sind in den Kommentaren im Beispiel auf der Vorlesungsseite beschreiben.