

Grundlagen der Praktischen Informatik

1. FUNKTIONALE PROGRAMMIERUNG

1.4 Alternativen

Betrachten wir folgende Definition des Betrags einer ganzen Zahl.

Um diese Definition umzusetzen, bietet sich ein bedingter Ausdruck an.

if *[test]* **then** *[expr_if]* **else** *[expr_else]*

Anhängig vom *[test]* (ein Ausdruck der nach `Bool` ausgewertet wird) nimmt der Ausdruck für

- *[test]* `== True` den Wert von *[expr_if]* an.
- *[test]* `== False` den Wert von *[expr_else]* an.

```
absolute :: Int -> Int
absolute x = if x < 0 then -x else x
```