*Aufgabe 1:*

*Aufgabe 2:*

User Story 1:

**Name** Ausgabe in Präfixnotation

**Beschreibung** Es steht eine Methode zur Verfügung, mit der bereits eingelesene boolesche Ausdrücke als

Strings in Präfixnotation ausgegeben werden können.

**Akzeptanzkriterium** Boolesche Ausdrücke können korrekt in Präfixnotation ausgegeben werden. Zwischen den Variablen und Operatoren steht jeweils ein Leerzeichen, kein Leerzeichen steht am Anfang und Ende des Strings.

User Story 2:

**Name** Ausgabe in Infixnotation

**Beschreibung** Es steht eine Methode zur Verfügung, mit der bereits eingelesene boolesche Ausdrücke als Strings in Infixnotation ausgegeben werden können.

**Akzeptanzkriterium** Boolesche Ausdrücke können korrekt in Infixnotation mit optimaler Klammerung ausgegeben werden. Der innere Ausdruck einer Negation steht nur dann in Klammern, wenn sich um ein And oder Or handelt, ein Teilausdruck eines And steht nur dann in Klammern, wenn sich um ein Or handelt, Teilausdrücke eines Or stehen nie in Klammern. Um die Operatoren And und Or stehen Leerzeichen, nach Not steht kein Leerzeichen, ebenfalls nicht am Anfang und Ende des Strings, nach öffnenden oder vor schließenden Klammern.

User Story 1:

Erste Runde(Algorithmus): 5 Minuten

# boolesche Ausdrücke->String in Präfixnotation

# Zwischen den Variablen und Operatoren steht jeweils ein Leerzeichen,

#kein Leerzeichen steht am Anfang und Ende des Strings.

Zweite Runde (Implement): 15 Minuten

#And

#Or

#Not

#Var

#AkzeptanzTest

User Story 2:

Erste Runde(Algorithmus): 10 Minuten

# boolesche Ausdrücke->String in Infixnotation

mit optimaler Klammerung ausgegeben:

#And: ein Teilausdruck eines And steht nur dann in Klammern, wenn sich um ein Or handelt,

stehen Leerzeichen

#Or: stehen nie in Klammern, Leerzeichen

#Not: steht nur dann in Klammern ,wenn sich um ein And oder Or handelt, nach Not steht kein Leerzeichen

#Var:

Leerzeichen stehen nicht am Anfang und Ende des Strings, nach öffnenden oder vor schließenden Klammern.

Zweite Runde(Implement): 20 Minuten

#And

#Or

#Not

#Var

#AkzeptanzTest

*Aufgabe 3:*

a)



geschätzte Story Points von Story 2:



Es scheint nict realistic:

Bei Userstory 2 ist das Problem in Erste Runde viel mehr komplizierter als bei erstem(4 > 2)

Die Implementierung ist auch viel mehr Kompliziert und es gibt mehrere TestCase daran.

b)



Nein, ich brauche mehr als 5 Minuten.

*Aufgabe 4:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **User Story** | **Story Points** | **Geplant für Iteration** | **Tatsächliche Zeit** | **Kundenpriorität** |
| US 5 | 7 | 2 | 3 Tage |  |
| US 6 | 12 | 1 | Nicht abgeschlossen | 4 |
| US 8 | 4 | 2 | 2 Tage |  |
| US 10 | 6 | 2 | 3 Tage |  |
| US 11 | 7 | 2 | Nicht abgeschlossen | 5 |
| US 14 | 6 | 2 | 4 Tage |  |
| US 15 | 2 | 2 | 1 Tag |  |
| US 16 | 3 |  |  | 6 |
| US 17 | 9 |  |  | 1 |
| US 18 | 2 |  |  | 3 |
| US 19 | 14 |  |  | 7 |
| US 20 | 6 |  |  | 8 |
| US 21 | 7 |  |  | 2 |

a) bei zweiter Iteration:



Realiserte Velocity bei erster Iteration:



Realisertbaren Story Points bei dritter Iteration:

