

Lernatelier: Projektdokumentation

Naray

Datum	Version	Änderung	Autor
25.08.2021	0.0.1	Beginn der Arbeit	Naray
01.09.2021	0.0.2	Weiterführung und Erweiterung der Projektdokumentation	Naray
01.09.2021	0.0.3	Beginn der Realisierung	Naray
08.09.2021	0.0.4	Weiterführung der Realisierung und Beginn des Debuggings	Naray
15.09.2021	0.0.5	Behebung aller Bugs durch Rubberducking	
	1.0.0	Finale Version	Naray

1. Informieren

1.1 Ihr Projekt

Spiel --> zufällige Zahl raten

In diesem Spiel generiert der Computer eine zufällige Zahl und der Spieler muss Sie mit Hinweisen innert 10 Vermutungen erraten.

1.2 Quellen

[Listen Sie hier explizit alle Quellen, die Sie benutzt wollen, um sich in das Projekt einzuarbeiten. Aktualisieren Sie diesen Teil, wenn Sie die Quellen bearbeitet haben.]

- LA1200 Einarbeitung Programmierung

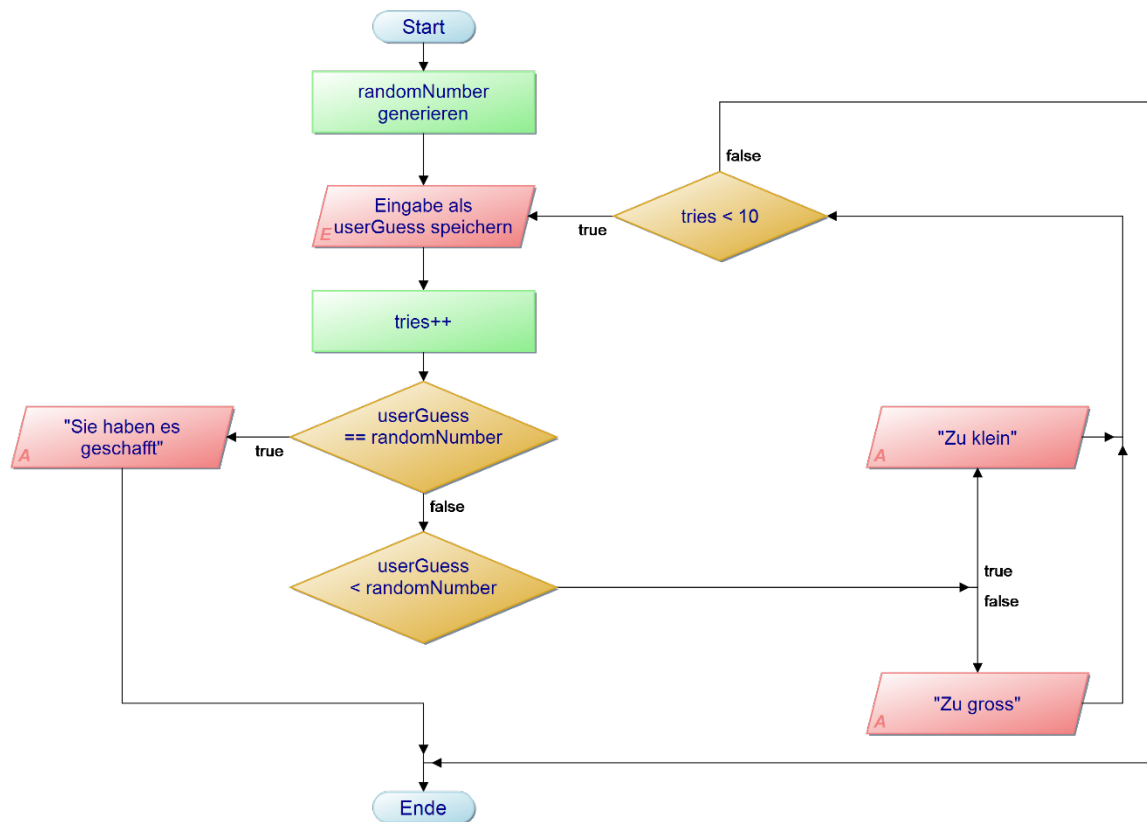
1.3 Anforderungen

Nummer	Muss / Kann?	Funktional? Qualität? Rand?	Beschreibung
1	muss	F	Der Computer speichert eine Zufallszahl zwischen 1 bis 100 als Geheimzahl.
2	muss	F	Der Benutzer kann Zahlen raten.
3	muss	Q	Für jede der geratenen Zahlen gibt der Computer einen Hinweis aus:
4	muss	Q	Die geratene Zahl ist niedriger als die Geheimzahl.
5	muss	Q	Die geratene Zahl ist grösser als die Geheimzahl.
6	muss	Q	Die Geheimzahl wurde erraten.
7	muss	R	Wenn die Geheimzahl erraten wurde, soll die Anzahl der Rateversuche ausgegeben werden.
8	muss	Q	Das Programm soll mit Fehleingaben umgehen oder sie vermeiden können.
9	muss	R	Der Text wird bei falschem Raten rot ausgegeben.

1.4 Diagramme

[Fügen Sie hier Anwendungsfall-Diagramme etc. ein.]

Auftrag 1200_2



1.5 Testfälle

[Erstellen Sie zu jeder Muss-Anforderung mindestens einen Testfall.]

Nummer	Vorbereitung	Eingabe	Erwartete Ausgabe
1.1	static void Main ist korrekt vorbereitet	Computer generiert zufällig eine Geheimzahl zwischen 1 und 100	Geheimzahl wird als Variable abgespeichert
2.1	Geheimzahl ist schon bekannt	Spieler gibt eine Zahl ein	Computer speichert diese Zahl als Variable ab
3.1	Geheimzahl und geratene Zahl sind bekannt	Computer vergleicht die Zahlen und gibt an, ob sie kleiner, grösser oder gleich ist	Wenn sie falsch ist, gibt der Computer einen Hinweis aus.
4.1	Geheimzahl und geratene Zahl sind bekannt	Zahl niedriger als Geheimzahl eingeben	"Die eingegebene Zahl ist kleiner als die Geheimzahl."
5.1	Geheimzahl und geratene Zahl sind bekannt	Zahl grösser als Geheimzahl eingeben	"Die eingegebene Zahl ist grösser als die Geheimzahl."
6.1	Geheimzahl und geratene Zahl sind bekannt	Zahl gleich zur Geheimzahl eingeben	"Sie haben es geschafft die Geheimzahl zu erraten!"
7.1	Zahl wurde erraten	Zahl gleich zur Geheimzahl eingeben	Anzahl versuche anzeigen
8.1	static void Main ist korrekt vorbereitet	Fehleingabe (alles ausser integer Zahl)	"Bitte versuchen sie es noch einmal mit einer natürlichen Zahl zwischen 1 und 100"
9.1	Es ist bereits ein Text vorhanden, welcher rot ausgegeben werden soll.	Eine Zahl, grösser oder kleiner als die Geheimzahl, wird eingegeben.	Die Ausgabe bei 4.1 oder 5.1 wird in Rot ausgegeben.

* Die Nummer hat das Format N.m, wobei N die Nummer der Anforderung ist, die mit dem Test abgedeckt wird, und m von 1 an fortlaufend durchnummeriert wird.

2. Planen

Nummer	Frist	Beschreibung	Zeit (geplant)
1.1	2.09.	Erstellung eines neuen Integers	45 min
2.1	2.09.	Der User kann eine Zahl eingeben	"
3.1	2.09.	Rückmeldung wird ausgegeben	"
4.1	2-8.09.	Die Rückmeldung wird korrekt ausgegeben	45 min
5.1	"	"	"
6.1	"	"	"
7.1	"	Benutzer wissen lassen, dass Zahl korrekt ist.	45 min
8.1	8.09.	Mit Fehleingaben umgehen	"
8.2	22.09.	Debugging und schöne Benutzeroberfläche	45 min
9.1	keine	Falls noch Zeit übrig, dann Färbung des Textes	45 min oder weniger

* Die Nummer hat das Format N.m, wobei N die Nummer der Anforderung ist, zu der das Arbeitspaket gehört, und m von 1 an fortlaufend durchnummeriert wird.

** Teilen Sie diesmal Ihre Anforderungen in 45-Minuten-Arbeitspakete ein

3. Entscheiden

[Dokumentieren Sie hier allfällige Entscheidungen, die Sie getroffen haben. Sonst lassen Sie dieses Kapitel leer.]

4. Realisieren

Nummer	Frist	Beschreibung	Zeit (geplant)	Zeit (effektiv)
1.1	2.09.	Erstellung eines neuen Integers	45 min	45 min
2.1	2.09.	Der User kann eine Zahl eingeben	"	"
3.1	2.09.	Rückmeldung wird ausgegeben	"	"
4.1	2-8.09.	Die Rückmeldung wird korrekt ausgegeben	45 min	45 min
5.1	"	"	"	"
6.1	"	"	"	"
7.1	"	Benutzer wissen lassen, dass Zahl korrekt ist.	45 min	45 min
8.1	8.09.	Mit Fehleingaben umgehen	"	"
8.2	22.09.	Debugging und schöne Benutzeroberfläche	45 min	45 min
9.1	keine	Falls noch Zeit übrig, dann Färbung des Textes	45 min oder weniger	15 min

[Übernehmen Sie Ihre Planung aus 2., und tragen Sie nach, wie lang Sie effektiv zur Bearbeitung der jeweiligen Arbeitspakete benötigt haben.]

5. Kontrollieren

5.1 Testprotokoll

Nummer	Datum	Resultat	Durchgeführt
1.1	15.09.2021	Geheimzahl wird als Variable abgespeichert	Naray
2.1	15.09.2021	Computer speichert diese Zahl als Variable ab	Naray
3.1	15.09.2021	Wenn sie falsch ist, gibt der Computer einen Hinweis aus.	Naray
4.1	15.09.2021	"Die eingegebene Zahl ist kleiner als die Geheimzahl."	Naray
5.1	15.09.2021	"Die eingegebene Zahl ist grösser als die Geheimzahl."	Naray
6.1	15.09.2021	"Sie haben es geschafft die Geheimzahl zu erraten!"	Naray
7.1	15.09.2021	Anzahl versuche anzeigen	Naray
8.1	15.09.2021	"Bitte versuchen sie es noch einmal mit einer natürlichen Zahl zwischen 1 und 100"	Naray
9.1	15.09.2021	Die Ausgabe bei 4.1 oder 5.1 wird in Rot ausgegeben.	Naray

[Vergessen Sie das Fazit aus dem Testprotokoll nicht!]

Beim Debuggen war alles korrekt ausgegeben. Alle Bugs wurden richtig behoben oder beseitigt, noch bevor ich die Testfälle durchgeführt habe.

6. Auswerten

[Listen Sie hier je mindestens einen Punkt, der gut gelaufen ist, und einen Punkt, der schlecht gelaufen ist – mit diesen starten Sie dann in Ihren Portfolio-Eintrag.]

Gut gelaufen:

Gut gelaufen ist die Flüssigkeit der Arbeit und am Ende die Sauberkeit des Programms. Alles läuft fehlerlos ab und das Programm ist benutzerfreundlich und Idiotensicher.

Schlecht gelaufen:

Der Code ist nicht der sauberste, da der "goto" Befehl verwendet wurde, um komplizierte Schleifen zu umgehen.