

## Portfolioprüfung – Werkstück A – Alternative 2

### 1 Aufgabe

Entwickeln und implementieren Sie ein Buzzword-Bingo-Spiel im Einzelspieler-Modus mit „grafischer Darstellung“ in der Shell.

Buzzword-Bingo ist ein Klassiker der IT-Geschichte und hilft dabei, die übermäßige Verwendung inhaltsleerer Schlagwörter (häufig Anglizismen) in Vorträgen bzw. Präsentationen sichtbar zu machen. Der Beschreibung von Buzzword-Bingo auf Wikipedia ist nichts hinzuzufügen:

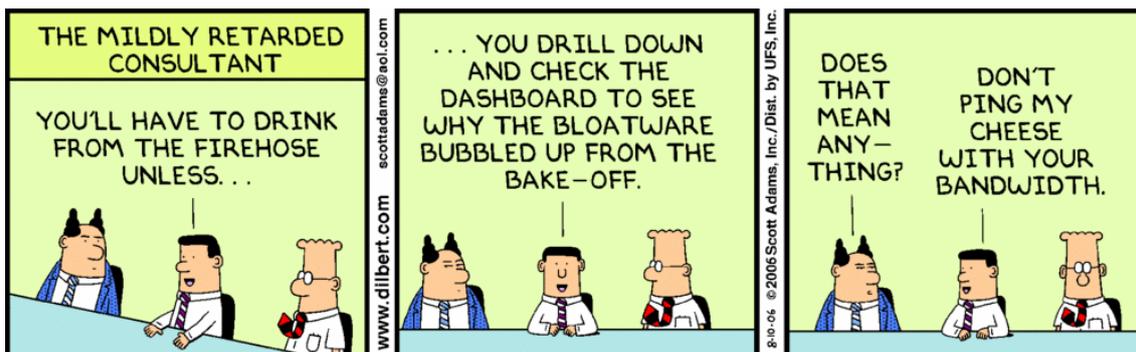
Buzzword-Bingo, in der späteren Verbreitung auch Bullshit-Bingo und Besprechungs-Bingo genannt, ist eine humoristische Variante des Bingo-Spiels, die die oft inhaltslose Verwendung zahlreicher Schlagwörter in Vorträgen, Präsentationen oder Besprechungen persifliert.

Statt Bingokarten mit Zahlen werden Karten mit Schlagwörtern (engl. buzzwords) benutzt. Im Gegensatz zum originalen Bingo, bei welchem die zu streichenden Zahlen aus einer Lostrommel gezogen werden, werden Wörter gestrichen, wenn sie genannt werden. Bei einer vollständig gefüllten Reihe, Spalte oder Diagonale soll der Spieler den Regeln nach aufstehen und *Bingo* oder auch, um die Inhaltsleere der Wortphrasen hervorzuheben, *Bullshit* rufen. Mit dem Spiel und diesem Ausruf wird gleichzeitig die übermäßige Verwendung oft inhaltsleerer Schlagwörter kritisiert.

Quelle: <http://de.wikipedia.org/wiki/Buzzword-Bingo>



Zudem sehenswert: <https://www.youtube.com/watch?v=d5zRe8wa4pM>



## 2 Anforderungen

- Benutzer sollen mit dem Spiel eine Bingokarte in der Shell generieren können.
- Die Anzahl der Felder in der Breite und Höhe der zu generierenden Bingokarte gibt der Benutzer per Kommandozeilenargument an. Also z.B. `-axis 5 -yaxis 5`
- Eine Export-Funktionalität (via Kommandozeilenargument!) ist nötig, um mehrere (verschiedene!) Bingokarten zu erzeugen, um diese ausdrucken zu können.
- Die Werte der Felder auf der Bingokarte sollen aus einer Textdatei eingelesen und per Zufall verteilt werden. Die Textdatei kann auch gerne mehr Wörter enthalten und es werden entsprechend so viele Wörter zufällig vom Buzzword-Bingo-Spiel aus der Datei importiert und auf der bzw. den Bingokarte(n) verteilt, wie es die definierte Höhe und Breite vorgibt.
- Mit der Tastatur (und evtl. auch mit der Maus) wählt der Benutzer im Einzelspieler-Modus einzelne Felder aus, um diese zu streichen (bzw. zu markieren).
- Hat der Benutzer eine komplette Spalte, Zeile oder Diagonale der Bingokarte an Feldern gestrichen bzw. markiert, gilt das Spiel als gewonnen, was entsprechend angezeigt wird. Das kann beispielsweise durch eine Laufschrift geschehen, durch ein Blinken oder durch ein Invertieren der Farben in der Shell, etc.
- **Entwickeln und implementieren Sie Ihre Lösung als Bash-Skript, als C-Programm oder als Python-Skript** als freie Software (Open Source) und verwenden Sie hierfür ein Code-Repository, z.B. bei GitHub.
- Für die „grafische Darstellung“ in der Shell verwenden Sie eine Bibliothek wie **ncurses**<sup>1 2</sup> (für C-Programme), **Termbox**<sup>3</sup> (für C-Programme oder Python-Skripte), **dialog**<sup>4 5 6</sup> (für Shell-Skripte) oder **Whiptail**<sup>7 8 9</sup> (für Shell-Skripte).
- Es soll möglich sein, das Spiel nur aus einer Shell heraus zu spielen. Es soll komplett in der Shell ablaufen!
- Bei 5x5 oder 7x7 Feldern bleibt das Feld in der Mitte üblicherweise frei (*Joker*).
- Der Quellcode soll durch Kommentare verständlich sein.
- Bearbeiten Sie die Aufgabe in Teams zu **3 Personen**.
- Schreiben Sie eine aussagekräftige und ansehnliche Dokumentation (Umfang: **8-10 Seiten**) über Ihre Lösung.

---

<sup>1</sup>[http://openbook.rheinwerk-verlag.de/linux\\_unix\\_programmierung/Kap13-002.htm](http://openbook.rheinwerk-verlag.de/linux_unix_programmierung/Kap13-002.htm)

<sup>2</sup>[https://de.wikibooks.org/wiki/Ncurses:\\_Grundlegendes](https://de.wikibooks.org/wiki/Ncurses:_Grundlegendes)

<sup>3</sup><https://github.com/nsf/termbox>

<sup>4</sup>[http://openbook.rheinwerk-verlag.de/shell\\_programmierung/shell\\_007\\_007.htm](http://openbook.rheinwerk-verlag.de/shell_programmierung/shell_007_007.htm)

<sup>5</sup><https://www.linux-community.de/ausgaben/linuxuser/2014/03/mehr-komfort/>

<sup>6</sup><https://linuxkurs.spline.de/Ressourcen/Folien/Linux-Kurs-7.pdf>

<sup>7</sup>[https://en.wikibooks.org/wiki/Bash\\_Shell\\_Scripting/Whiptail](https://en.wikibooks.org/wiki/Bash_Shell_Scripting/Whiptail)

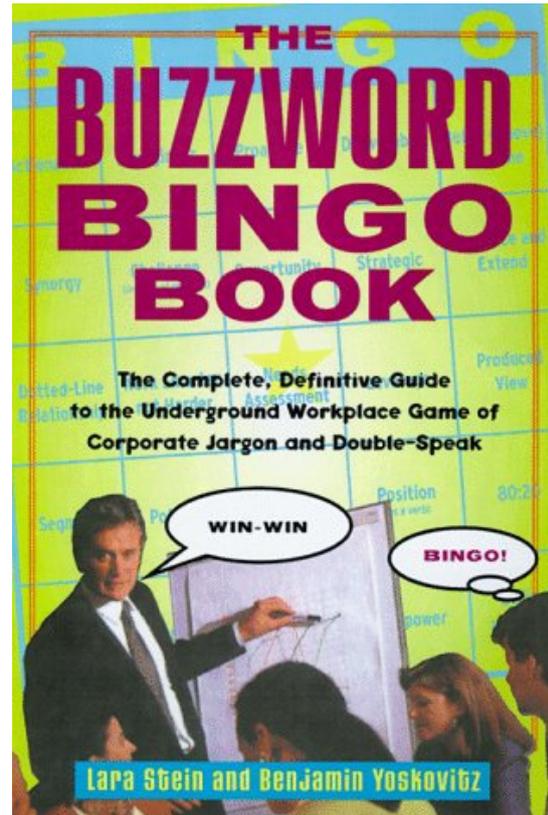
<sup>8</sup><https://saveriomiroddi.github.io/Shell-scripting-adventures-part-3/>

<sup>9</sup><https://www.dev-insider.de/dialogboxen-mit-whiptail-erstellen-a-860990/>

- Die Funktionalität der Lösung müssen Sie in der Übung demonstrieren. Bereiten Sie einen Vortrag mit Präsentationsfolien und eine Live-Demonstration (Umfang: **15-20 Minuten**) vor.

### 3 Einige Buzzwords zur Inspiration

- Synergie
- Rating
- Wert-schöpfend
- Benefits
- Ergebnisorientiert
- Nachhaltig
- Hut aufhaben
- Visionen
- Zielführend
- Global Player
- Rund sein
- Szenario
- Diversity
- Corporate Identitiy
- Fokussieren
- Target
- Benchmark
- Herausforderung(en)/Challenges
- Gadget
- Synergie
- Value
- Smart
- Web 2.0 oder 3.0
- Qualität
- Big Picture
- Revolution
- Pro-aktiv
- Blog
- Community
- Social Media
- SOA
- Skalierbar
- Return on Invest (ROI)
- Wissenstransfer
- Best Practice
- Positionierung/Positionieren
- Committen
- Geforwarded
- Dissemination
- Skills
- Gap
- Follower
- Win-Win



Bildquelle: <http://www.brokenwire.net>