

Praktische Klassenarbeiten in Programmieren

Beitrag von „Drew“ vom 9. Mai 2006 23:38

Forsch:

Vielen Dank für Deine Anregungen. Ja, den Lösungsansatz und die Planung kann ich auch bewerten. Mein grosses Problem ist, dass ich zwei verschiedene Noten für Theorie und Praxis bilden muss (und zwar mit zwei verschiedenen Klassenarbeiten). Meist lasse ich die Schüler in der Theorie-[Klassenarbeit](#) das Planen, was sie in der Praxis-[Klassenarbeit](#) dann umsetzen (coden und testen) sollen. Auch die Planung in die Praxis-[Klassenarbeit](#) mit aufzunehmen führt leider zu enormen Zeitproblemen bei den Schülern, da die Programme für die zur Verfügung stehenden 60 min. mittlerweile schon recht aufwendig geworden sind (Aktuell: OOP Design Patterns in Java).

Ausserdem habe ich bei früheren Klassenarbeiten festgestellt, dass eine zu starke Gewichtung des Lösungsansatzes dazu führt, dass der Schüler eine recht gute Note für ein schlechtes "Outcome" im Ganzen betrachtet bekommt. (Deshalb habe ich u.a. den 50% Punktabzug bei nicht-kompilierbaren Programmen eingeführt).

Ich fände es nicht schlecht, wenn bei uns Theorie und Praxis einfach zu einer gemeinsamen Note führen würde.

Es sollte einfach nicht passieren, dass ein Schüler zwischen Theorie und Praxis zu starke Abweichungen zeigt. (Was allerdings für die Aufnahme der Planung in den Praxis-Teil spricht!). Aber was mache ich dann in Theorie?

Für mich als Informatiker ist Programmieren reine Theorie. Jemand, der seine Stärken lediglich im Reinhacken von Code hat ist heute in der Industrie leicht ersetzbar. (Hab' mal wo gearbeitet, da hat eine zuvor arbeitslose Ärztin Java programmiert. Frag' nicht wie der Code aussah, aber früher oder später und mit ein bisschen nachhelfen meinerseits hat's dann sogar funktioniert).

Gibt es bei Deinen Programmierklassen auch getrennte Noten oder ist das nur bei uns in BW und an bestimmten Schularten üblich?

-- Drew