

→ C O in gith	.b.com/ace-lectures/mutation-dem	0	ਸੇ U 🙂 🖉 🤻 🌚 :	Additional
ace-lectures/mu	tation-demo	⊙ Watch + 1	☆ Star 2 및 Fork 1	Matorial
<> Code () Issues	1) Pull requests 1 💿 Act	ions 🗉 Projects 🖽 W	ki 🛈 Security 🚥	Material
P master -	Go to file	Add file • 👱 Code •	About 🛞	
mosser Update REA	DME.md =	2 hours ago 🔞 6	No description, website, or topics provided.	Code
docs	first version	yesterday	Readme	
plantumi	first version	yesterday		Running exampl
poker-hands-kata	first version	yesterday	Releases	
nsrc/main/java	first version	yesterday	No releases published Create a new release	Design models
🗅 .gitignore	first version	yesterday		
README.md	Update README.md	2 hours ago	Packages	
🗅 clean.sh	first version	yesterday	Sébastien Mosser @	petitroll - 14 mars
🗅 pom.xml	first version	yesterday	Interested in mutation a simple mutation test	testing? Here is a small proof of concept to cre ing framework (for pedagonical purpose) in ~3
prof-x.sh	first version	yesterday	lines of Java and 60 lin	nes of Bash.







Flement	Missed Instructions #	Cove	Missed Branches	Cove	
@ Rank.Pair		68%		66%	
G Rank, TwoPair		69%		50%	
G Card	-	66%	1	90%	
G Rank.HighCard		57%		50%	
G Rank.FourOfKind	-	57%		50%	
G Rank.Straight		57%	-	50%	
G TwoCards	-	69%		50%	
<u>Cards</u>		89%	-	78%	
G Rank.ThreeOfKind	-	71%		50%	
G Winner		74%		50%	
G Rank.Flush	-	72%		50%	
G Rank.StraightFlush		70%		50%	
G RankFormatter		94%		75%	
ThreeCards		71%		n/a	

** Generated 78 neutrins. Killed 66 (785) ** See State	103151133/index.html
Prit Test Coverage Report Project Summary Number of Line Coverage Mutation Coverage Test Strength	103151133/index.html
Pit Test Coverage Report [100] Total time: 26.488 s roject Summary [100] Finished et: 2821-83-15T11:34:16-96:00 Number of Classe Line Coverage Mutation Coverage Test Strength	103151133/index.html
The Less Coverage Report [[mo] Finished et: 2021-83-15711:34:16-04:80 roject Summary [[mo] Finished et: 2021-83-15711:34:16-04:80 Number of Classe Line Coverage Mutation Coverage Test Strength [[mo] Finished et: Strength	103151133/index.html
roject Summary bosts#104() poker-hands-kets % open target/pit-reports/2021033 Number of Line Coverage Mutation Coverage Test Strength	103151133/index.html
Number of Line Coverage Mutation Coverage Test Strength	
Number of Line Coverage Mutation Coverage Test Strength	
3 92% 73% 88%	
312/338 196/270 196/223	
reakdown by Package	
Number of the Group Matrix Group Test Street	
Name Classee Line Coverage Mutation Coverage Test Strength	







Which abstractions do we need for mutations?



- Find the spots where a mutation can be used
- Rewrite a given program at a given spot
- Trace the mutations
- Read the **configuration** (e.g., mutation level)

















Perspectives: Software Engineering Challenges!

ON SOFTWARE END



· Research:

- Mutant selection
- Mutant equivalence
- Traceability

• Engineering

- Reporting
- Performances

An Analysis and Survey of the Development of Mutation Testing

Yue Jia, Student Member, IEEE, and Mark Harman, Member, IEEE

ING, VOL. 37, NO. 5, SEPT

Abstract—Mutation Texting is a faul based otherwise testing technique that has been wiskly studied for over three discusts. The Bitracture on Mutation Texting has contributed a set of approaches, toxit, development, and empirical results. This paper provides a program advectory provided testing. The space advector the treaded streament is readed streament to readed streament the readed streament to readed streament the readed streament testing testing to advect testing te

dex Terms—Mutation testing, survey.

Thanks for your attention!

