



TO DO : Open a web browser (firefox?)
You should go to my home page and click on the link:
“Semaine Transverse : the problem with e-voting”

Home Page for J. Paul Gibson (B.Sc, Ph.D, HDR)

Département - INFormatic

Location: Bureau D311, Le département INFormatic, Telecom SudParis (TSP), 9 rue Charles Fourier, 91011 Évry cedex, FRANCE

Research: L'Unité Mixte de Recherche SAMOVAX (Services répartis, Architectures, MOdélisation, Validation, Administration des Réseaux) UMR 5157 INT CNRS Research Laboratory, Equipe METHODES (Méthodes et modèles pour les réseaux),

skype: jpaulgibson, @JPaulGibson
 email: paul.gibson@telecom-sudparis.eu, Phone/Fax: +33(0)1-6076 (4477/4711), Contact/Annuaire TSP

A recent CV is available in English or French (français).

My Web Pages

- Research
 - Publications and Invited Presentations
 - Conferences/Workshops: committees and reviewing
 - Supervision: PhD, MSc and BSc.
 - Evaluation/Reviewer: journals, projects and theses.
- Current Teaching (2015/16)
 - Undergraduate TSP -
 - Semaine Transverse - The problem(s) with e-voting
 - CSC3101 - Algorithmique et langage de programmation
 - CSC3502 - Projet Informatique
 - CSC4102 - Introduction au génie logiciel pour applications orientées objet
 - CSC4504 - Langages formels et applications
 - CSC4522 - Projet système d'information : réalisation et déploiement
 - CSC5524 - Industrialisation d'un projet système d'information: software quality, metrics, tests, process improvement
 - Postgraduate TSP -
 - MSc Computer & Communication Networks (CCN):
 - CSC7003 - Foundations of Software Engineering
 - CSC7203 - Object Oriented Computing and Distributed Systems
 - CSC7336 - Software Engineering For Smart Devices
 - Formation Continue (Further Education) - Steria Architect Academy
- Personal Pages

External Links

- ANR research project
- CS Unplugged
- Annals of telecommunications (Springer Journal)
- 7TH INTERNATIONAL SYMPOSIUM ON LEVERAGING APPLICATIONS OF FORMAL METHODS, VERIFICATION AND VALIDATION
- SHONAN MEETING Implicit and Explicit Semantics Integration In Proof Based Developments Of Discrete Systems.
- Special Issue Journal - The role of telecommunications in E-voting





<http://www-public.tem-tsp.eu/~gibson/Teaching/SemaineTransverse/E-voting-problems/>


MINES TELECOM TELECOM
INSTITUT Mines-Télécom SudParis

Teaching-SemaineTransverse/E-voting-problems for Dr J. Paul Gibson, INformatique (INF), Telecom SudParis, France.


Semaine Transverse - E-voting Problems

In this short TP we will be looking at some of the problems that are associated with e-voting machines. You will be presented with some on-line voting machines and you will have to experiment with them to see if they function as required.





Scratch Machines On-Line:  machine 1,  machine 2,  machine 3,  machine 4

The machines are written using Scratch  a simple graphical programming language for teaching beginners to programming (often, but not always, young children)



Lecture Slides

E-voting Problems  slides,

Additional reading material

 *A Review of E-voting: the past, present and future*, J Paul Gibson, Robert Krimmer, Vanessa Teague and Julia Polmares. Published in Springer Annals of Telecommunications, volume=71, number=7, pages=279--286, July 2016
 PDF  Springer Link  10.1007/s12243-016-0525-8

Useful Links

  Scratch Programming

URL: <http://www-public.telecom-sudparis.eu/~gibson/Teaching/SemaineTransverse/E-voting-problems/> Last Revision: 15th September 2016 Contact: paul.gibson@telecom-sudparis.eu

TO DO : click on the first Scratch machine1

TO DO: wait until the project has loaded

The screenshot shows the Scratch web interface. At the top, there is a navigation bar with 'Scratch', 'Create', 'Explore', 'Discuss', 'About', 'Help', a search bar, 'Join Scratch', and 'Sign in'. Below this, the project title 'VotingMachine1' by JPaulGibson is displayed. To the right of the title, there are statistics for '0 scripts' and '0 sprites', and a 'See inside' button. A 'DRAFT' label is also present. The main stage area is mostly greyed out, with a white dialog box in the center that says 'Loading project...'. The right sidebar contains two sections: 'Instructions' and 'Notes and Credits'. The 'Instructions' section contains the following text: 'This is a simple example of an e-voting machine. It is used in my teaching as an introduction to computer science and software engineering. It supports discussion on : e-voting system requirements'. The 'Notes and Credits' section is currently empty. At the bottom of the sidebar, there is a 'work-in-progress' label, a copyright notice '© Shared: 15 Sep 2016', and a modification date 'Modified: 15 Sep 2016'.

TO DO: run the machine by pressing on the **green flag**

VotingMachine1
by JPaulGibson

16 scripts
3 sprites

DRAFT See inside

Instructions

This is a simple example of an e-voting machine.

It is used in my teaching as an introduction to computer science and software engineering.

It supports discussion on :

e-voting system requirements
..

Notes and Credits

work-in-progress

© Shared: 15 Sep 2016 Modified: 15 Sep 2016

You can go to full screen mode

Electronic Voting Terminal

To open the voting process
requires special authorisation

What's your authorisation to start the election



Can you guess the authorisation password?

It is surprising how many e-voting systems are not secure

Electronic Voting Terminal

To open the voting process
requires special authorisation

How many voters are on the electoral list?

Election configuration is usually more complex, but even in this simple case the administrators can make mistakes


TO DO : Test which values are considered valid

TimeLeftToVote **60**

Electronic Voting Terminal

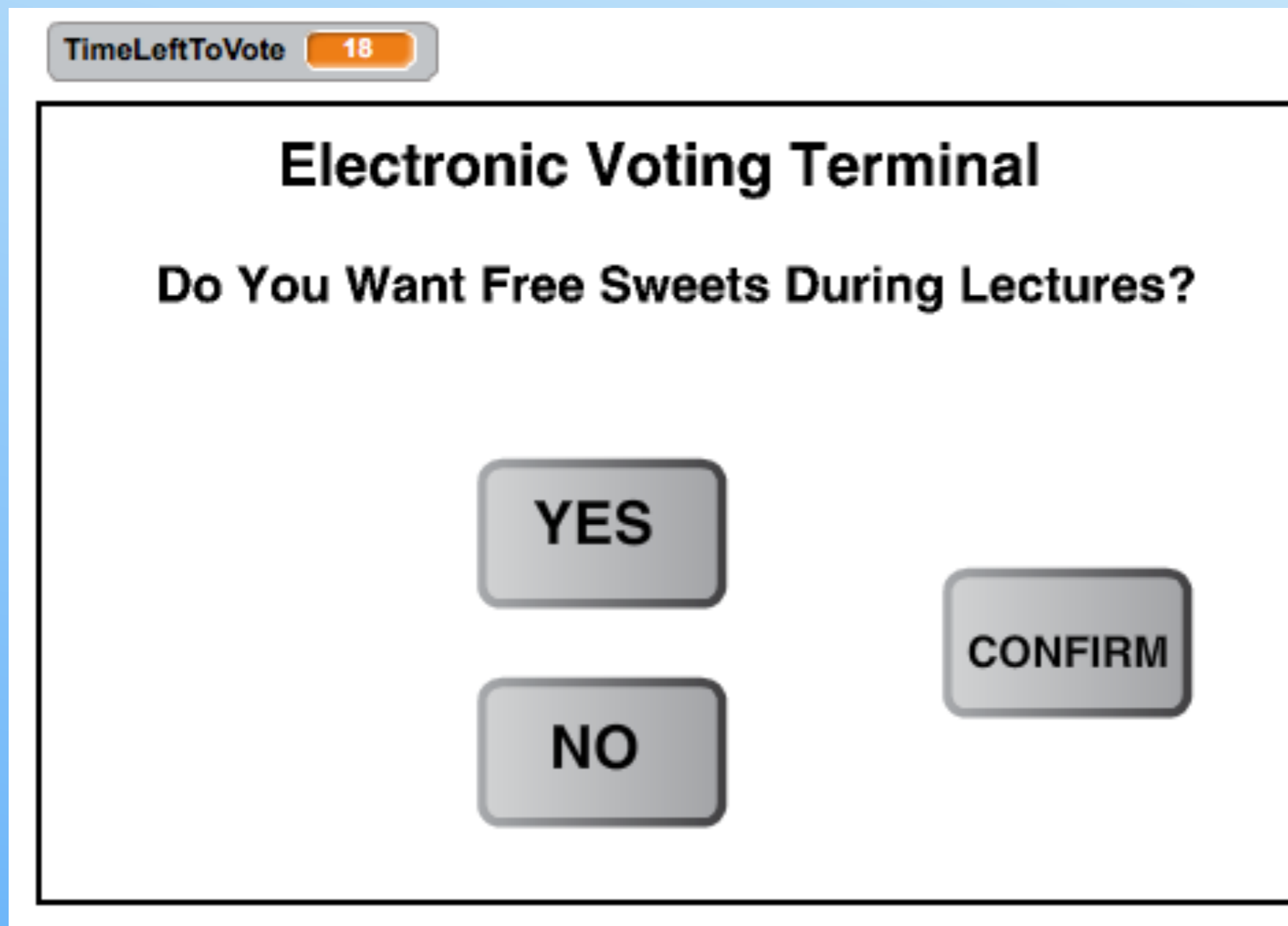
Please identify yourself as a registered voter

Authorisation code?

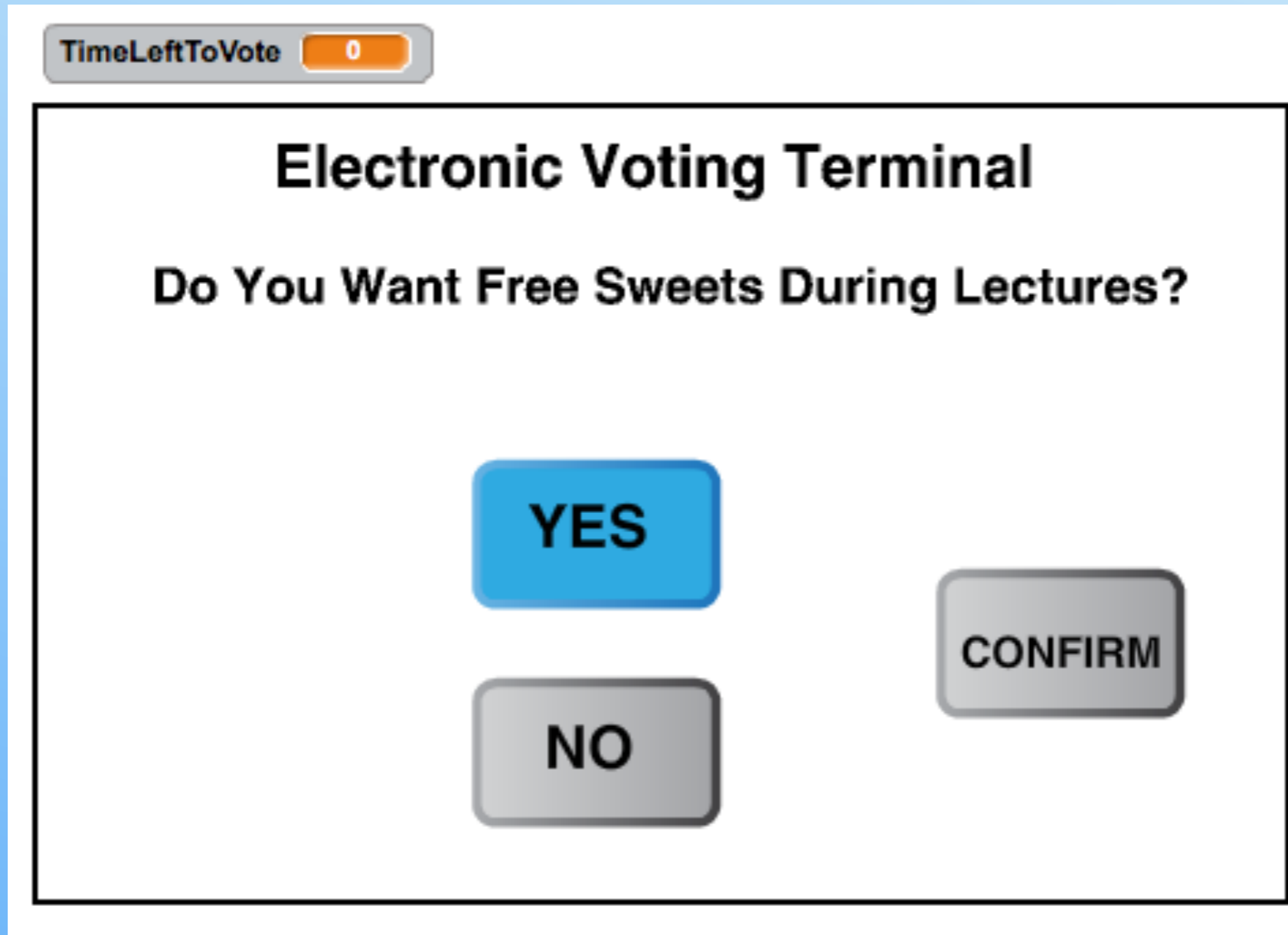


Voter authentication is usually complex, but here each voter has a unique ID number between 0 and n-1

Question : is this sort of code a good idea?



TO DO : Test if the interface works *correctly*



QUESTION : what happens if the time runs out in the middle of voting?

Electronic Voting Terminal

The Election Is Closed

The result is :

count_yes 1

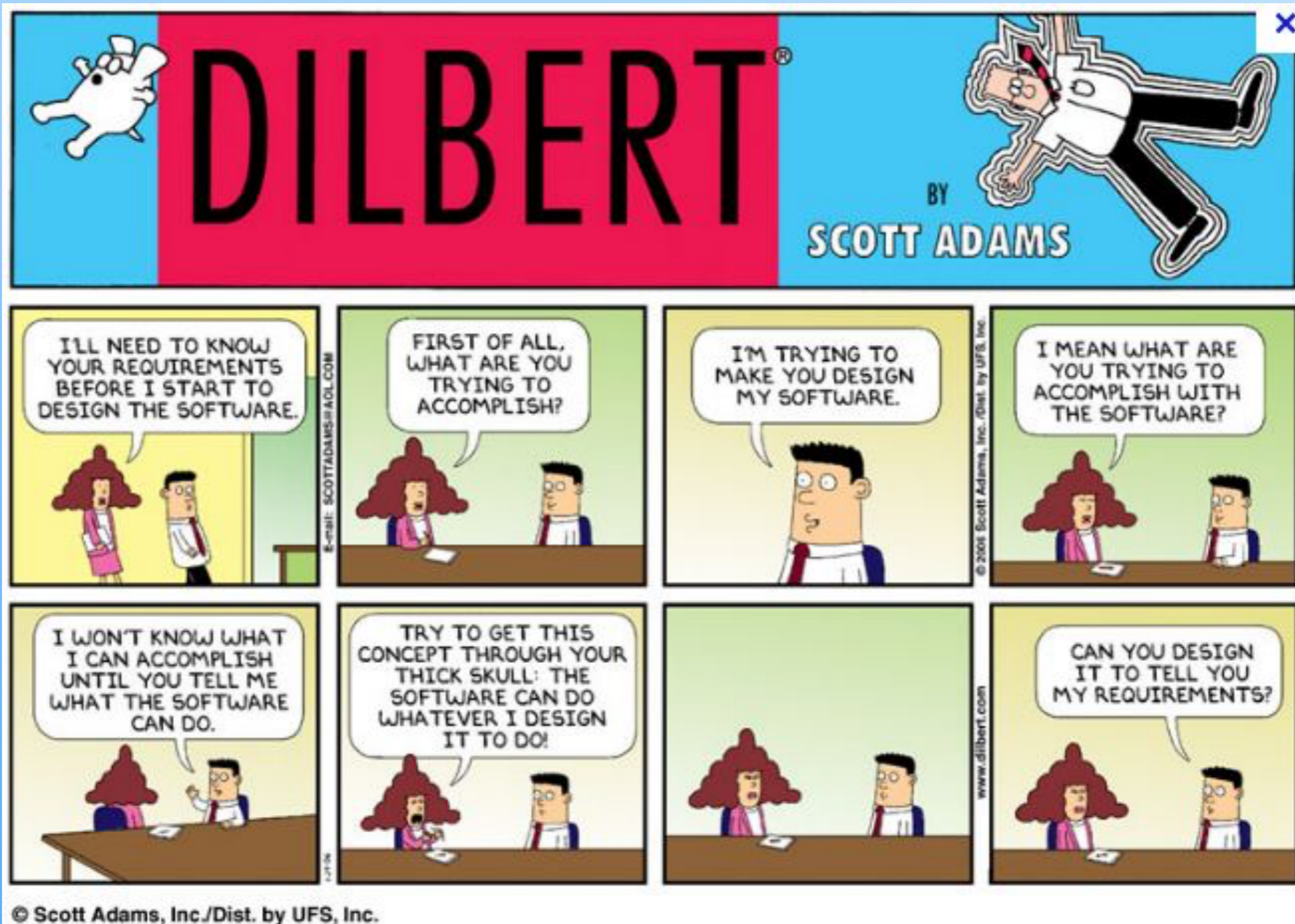
count_no 0

Does the election count the votes *correctly*?

Is the machine *acceptable*?

E-voting system requirements

What requirements do you think a voting system should meet?



E-voting system requirements

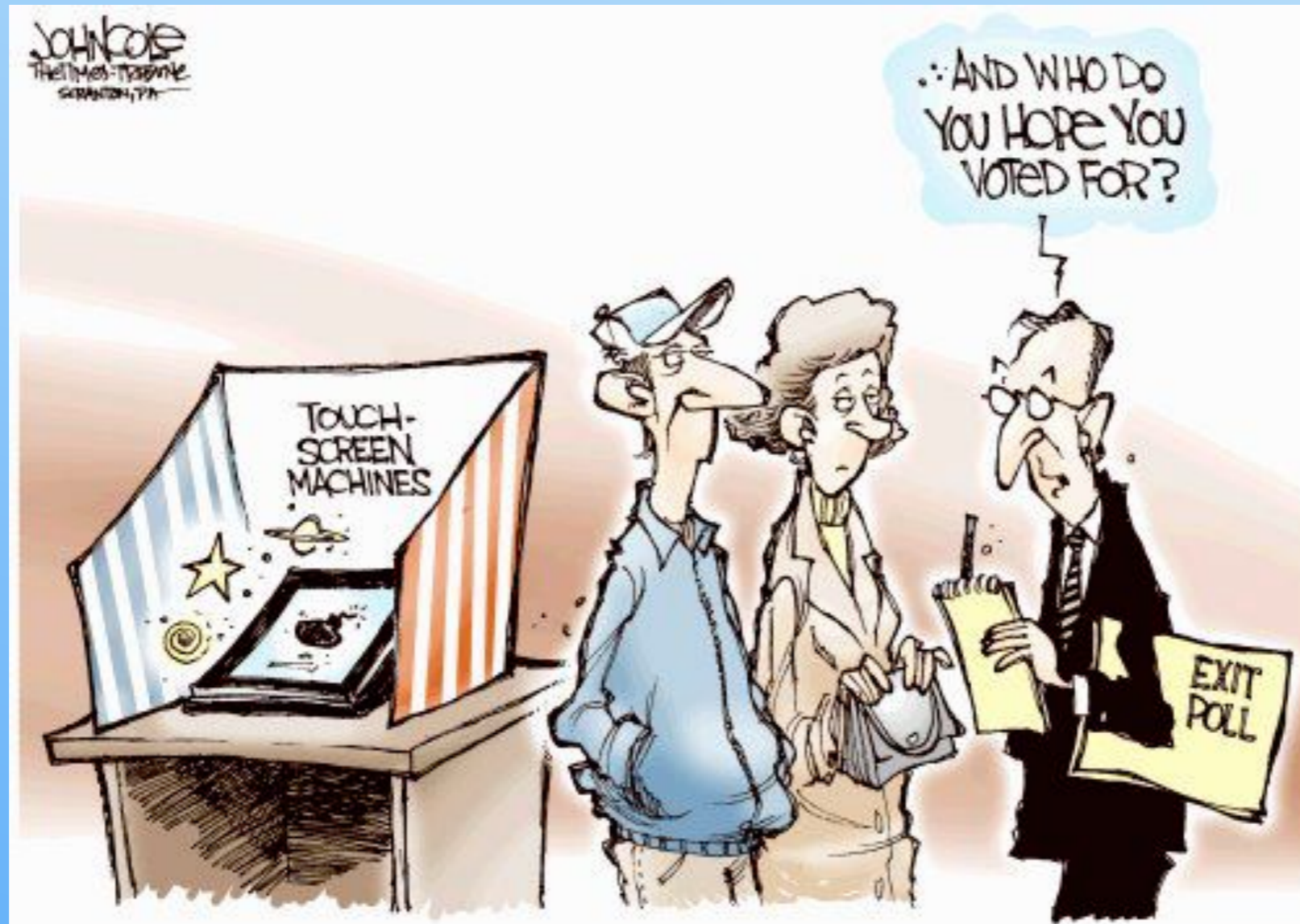
What requirements do you think a voting system should meet?



TRUSTWORTHY - TRUSTED???

E-voting system requirements

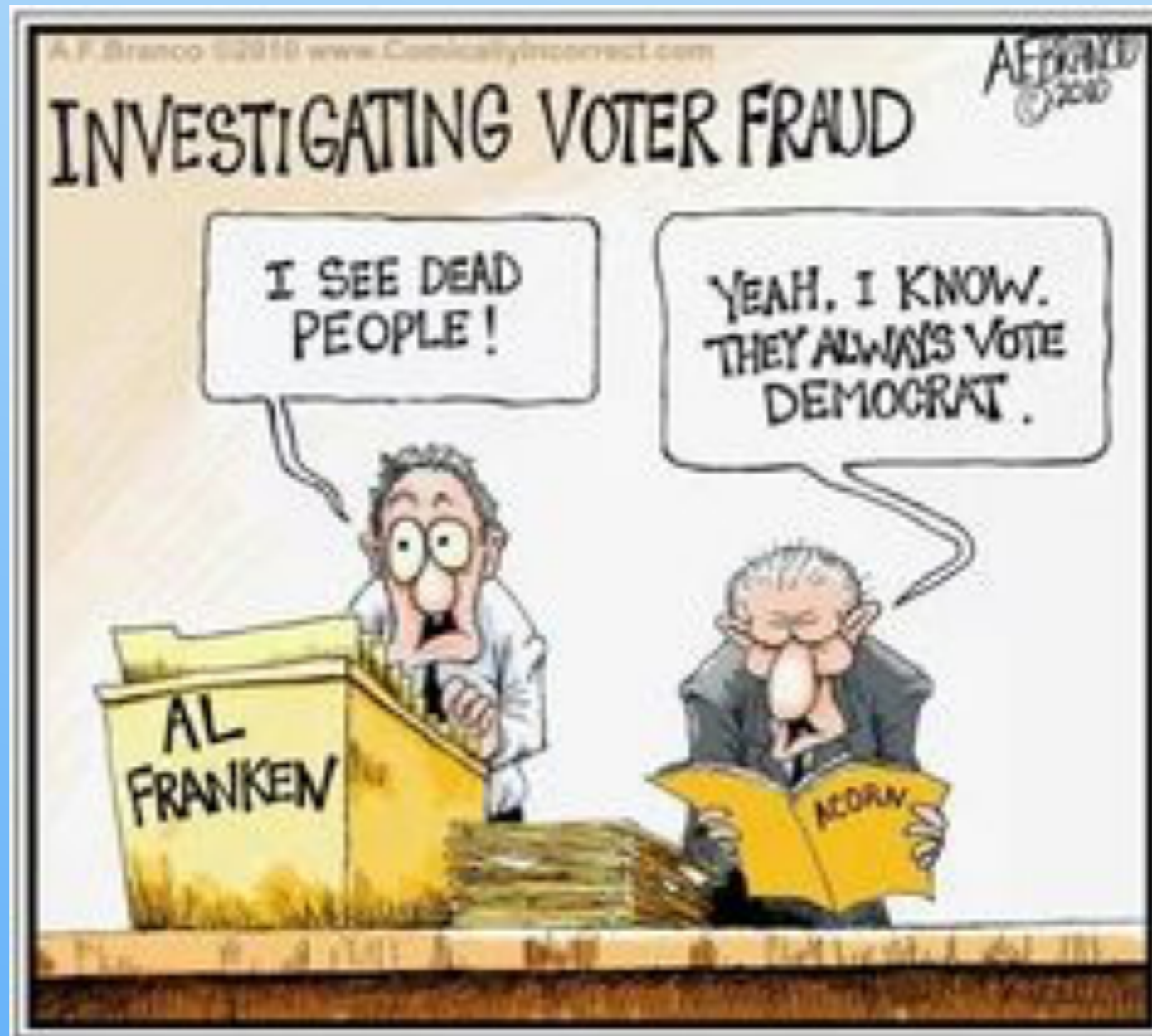
What requirements do you think a voting system should meet?



RELIABLE - DEPENDABLE???

E-voting system requirements

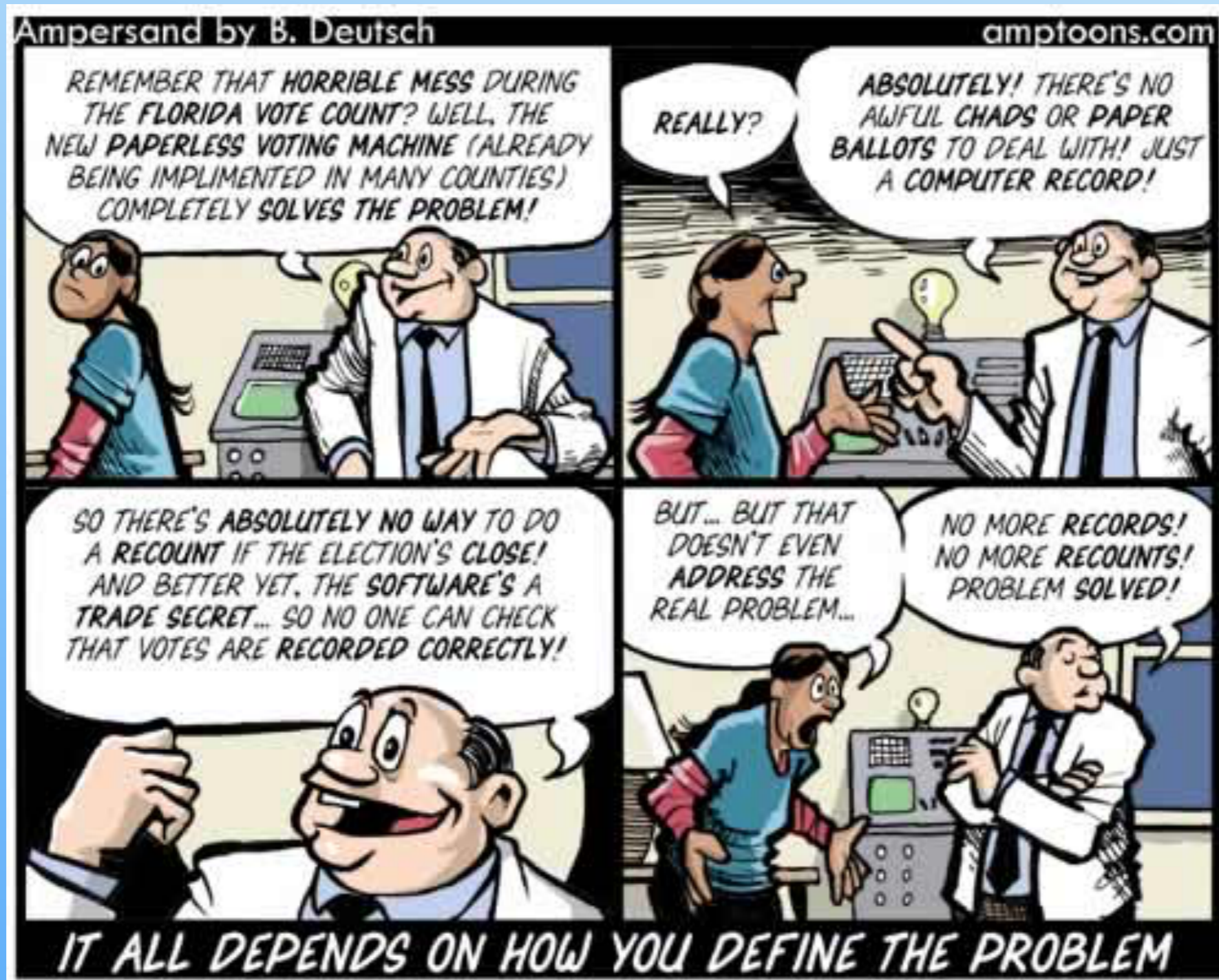
What requirements do you think a voting system should meet?



SAFE AGAINST FRAUD???

E-voting system requirements

What requirements do you think a voting system should meet?

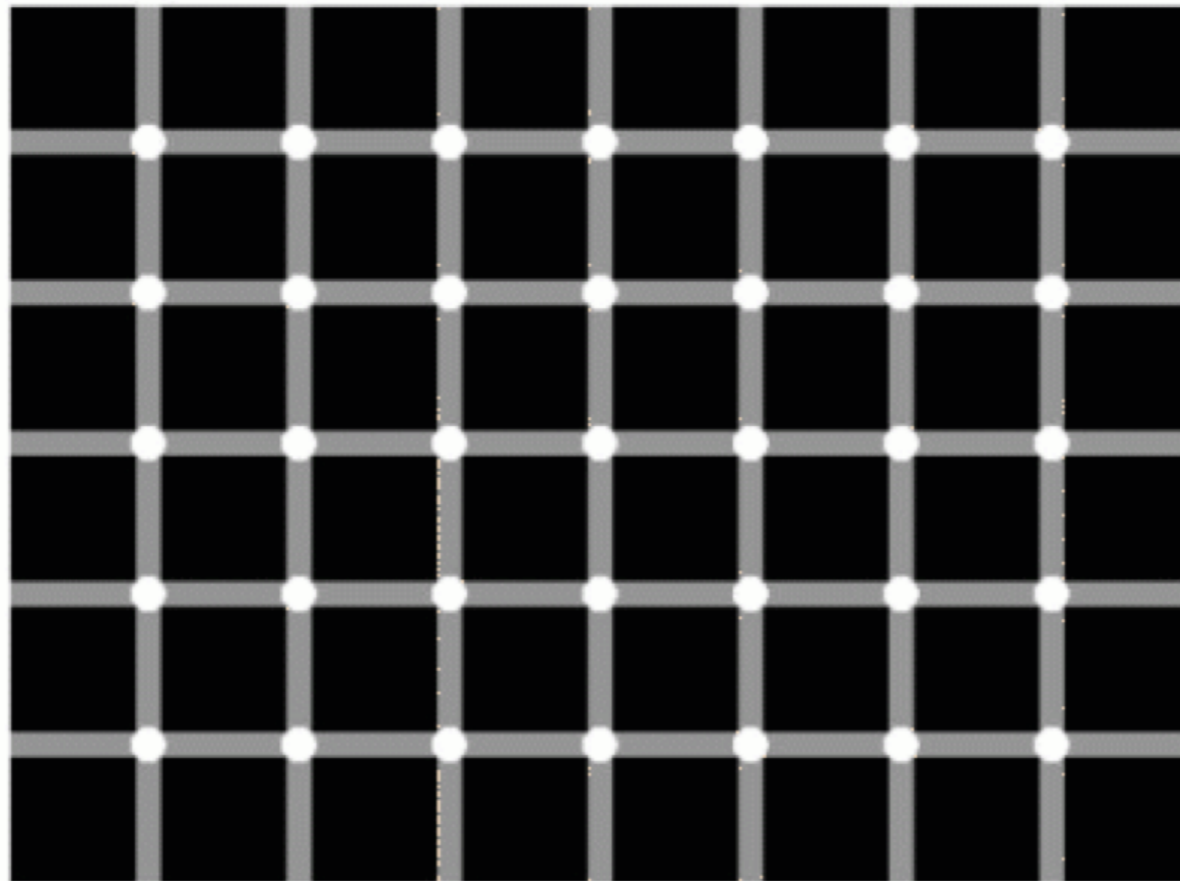


AUDITED??

E-voting system requirements

What requirements do you think a voting system should meet?

Florida Election Recount



Count and total black dots for Al Gore and white dots for George Bush. Recount to confirm

RECOUNTABLE ??

E-voting system requirements

What requirements do you think a voting system should meet?



RECEIPT-FREE ??

E-voting system requirements

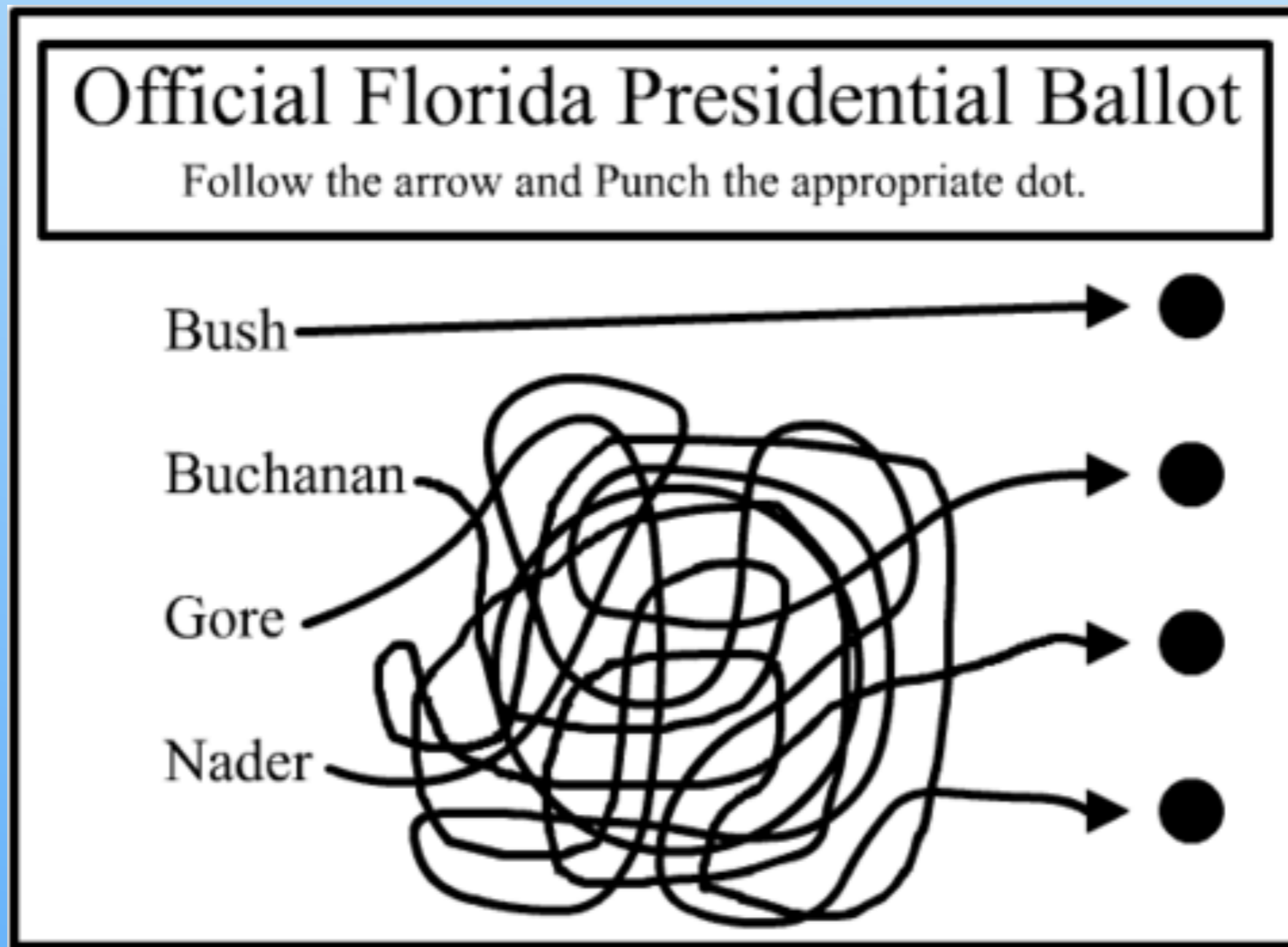
What requirements do you think a voting system should meet?



ON A NETWORK??

E-voting system requirements

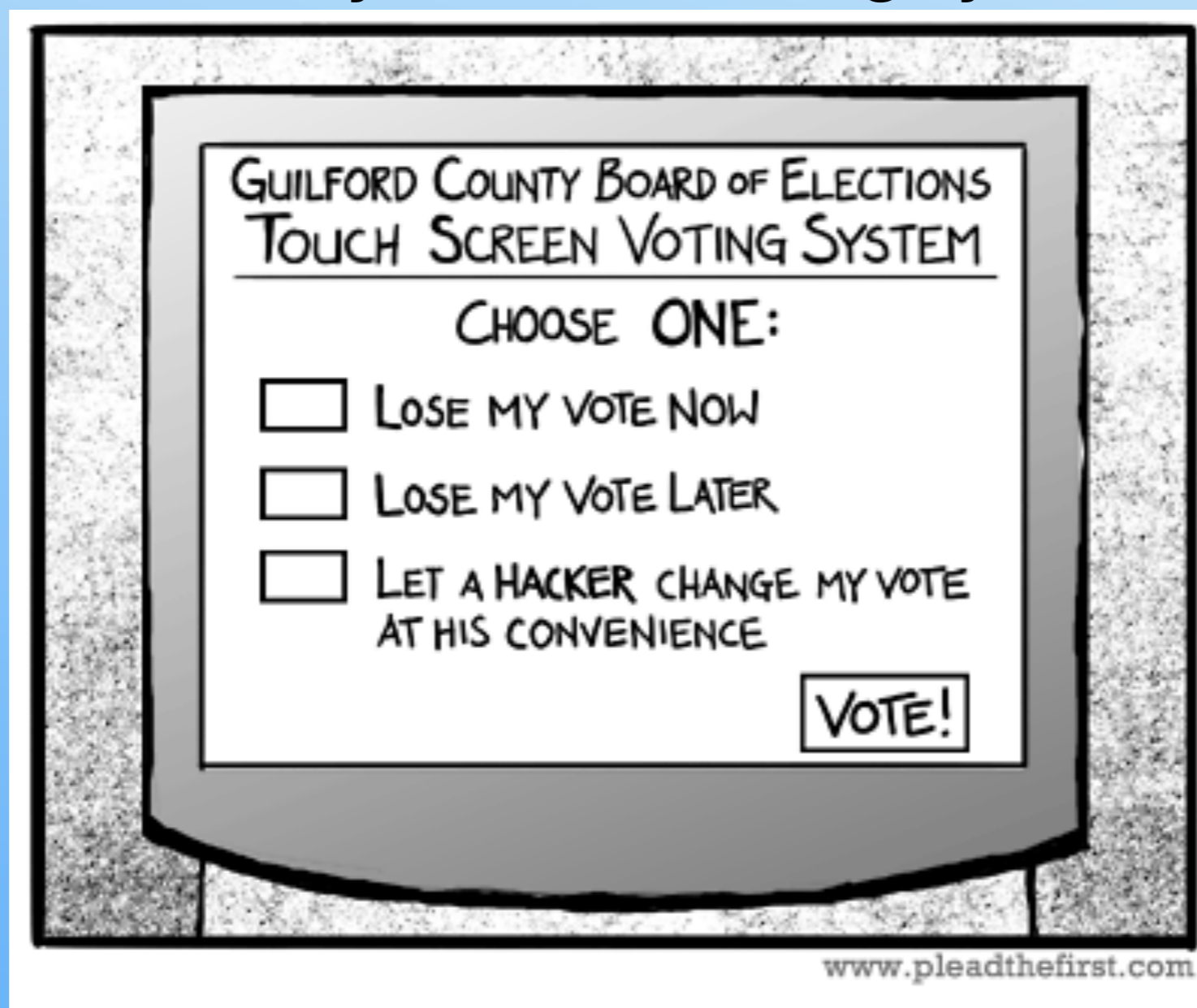
What requirements do you think a voting system should meet?



USABLE??

E-voting system requirements

What requirements do you think a voting system should meet?



SECURE AGAINST ATTACKS

E-voting system requirements

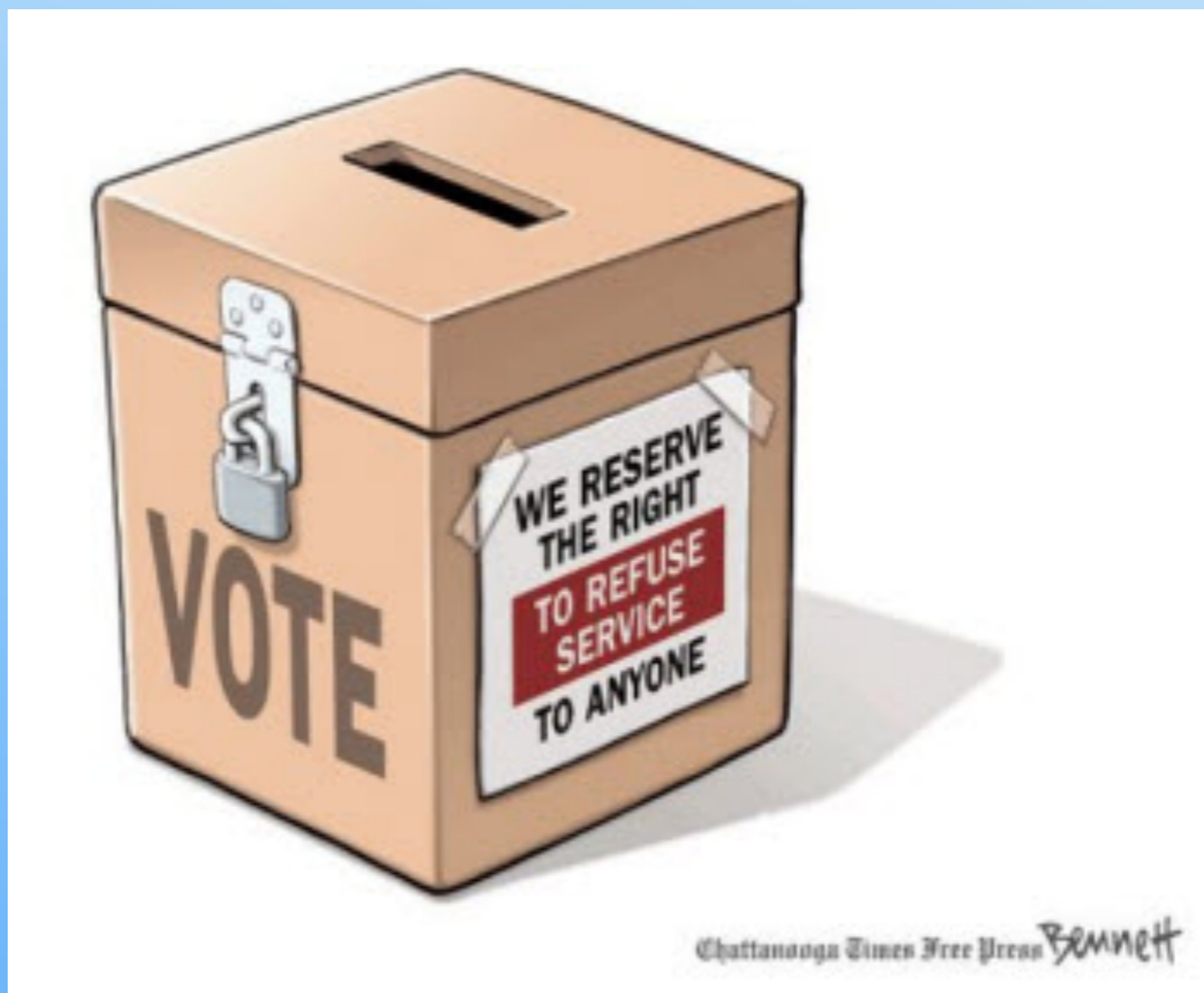
What requirements do you think a voting system should meet?



OPEN / TRANSPARENT ??

E-voting system requirements

What requirements do you think a voting system should meet?



ACCESSIBLE

E-voting system requirements

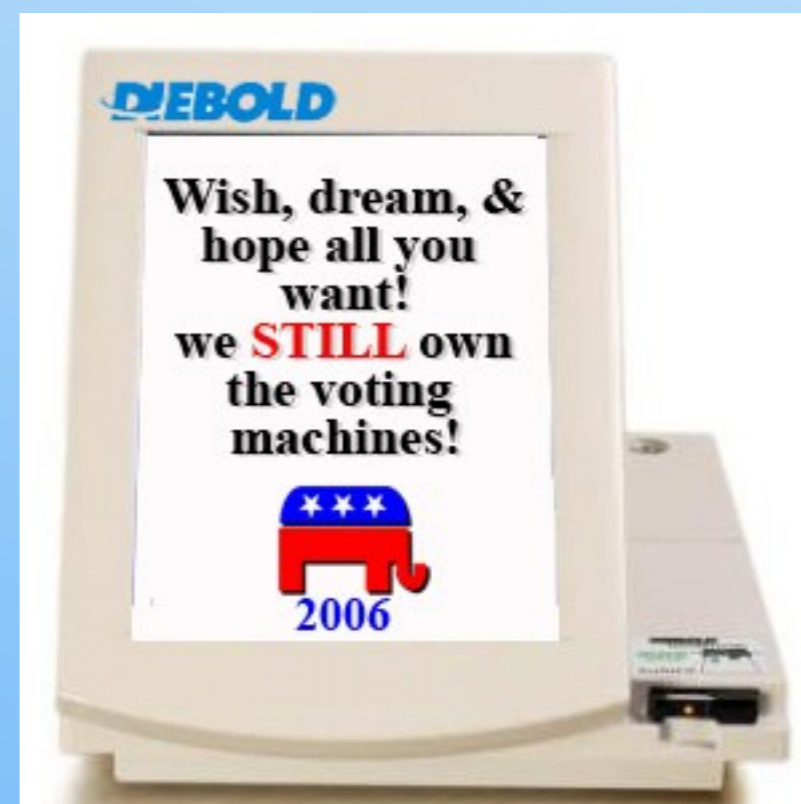
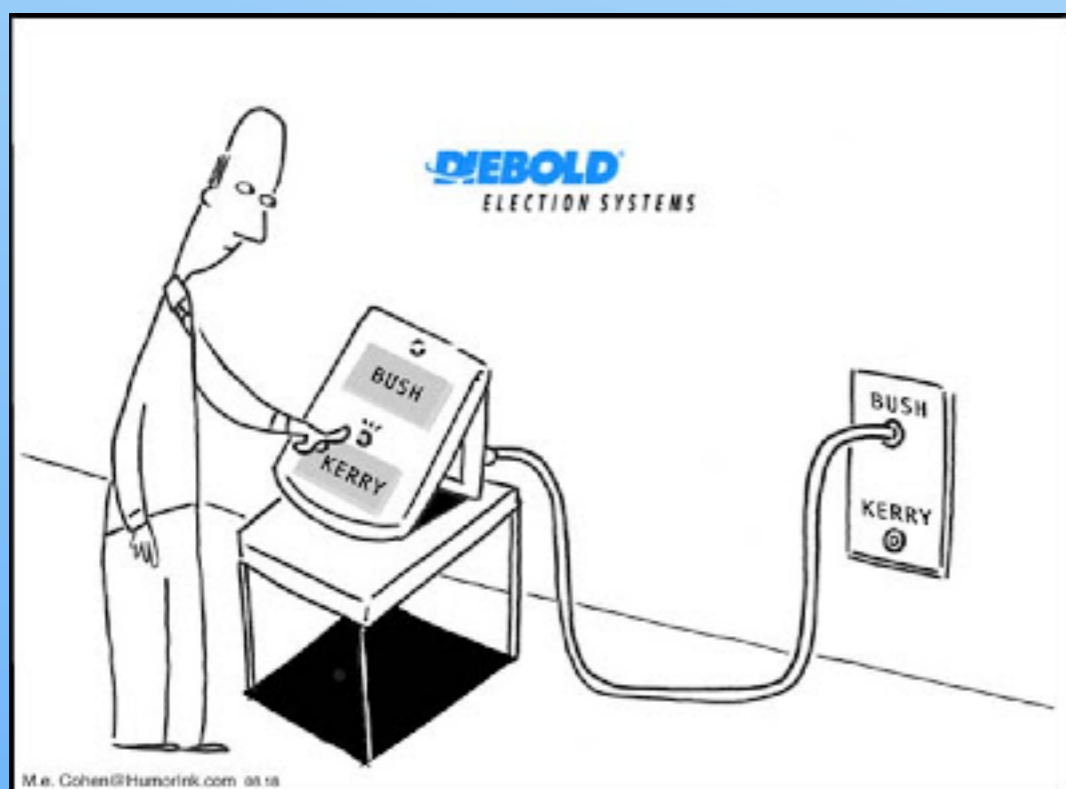
What requirements do you think a voting system should meet?



(FULLY) ELECTRONIQUE?

E-voting system requirements

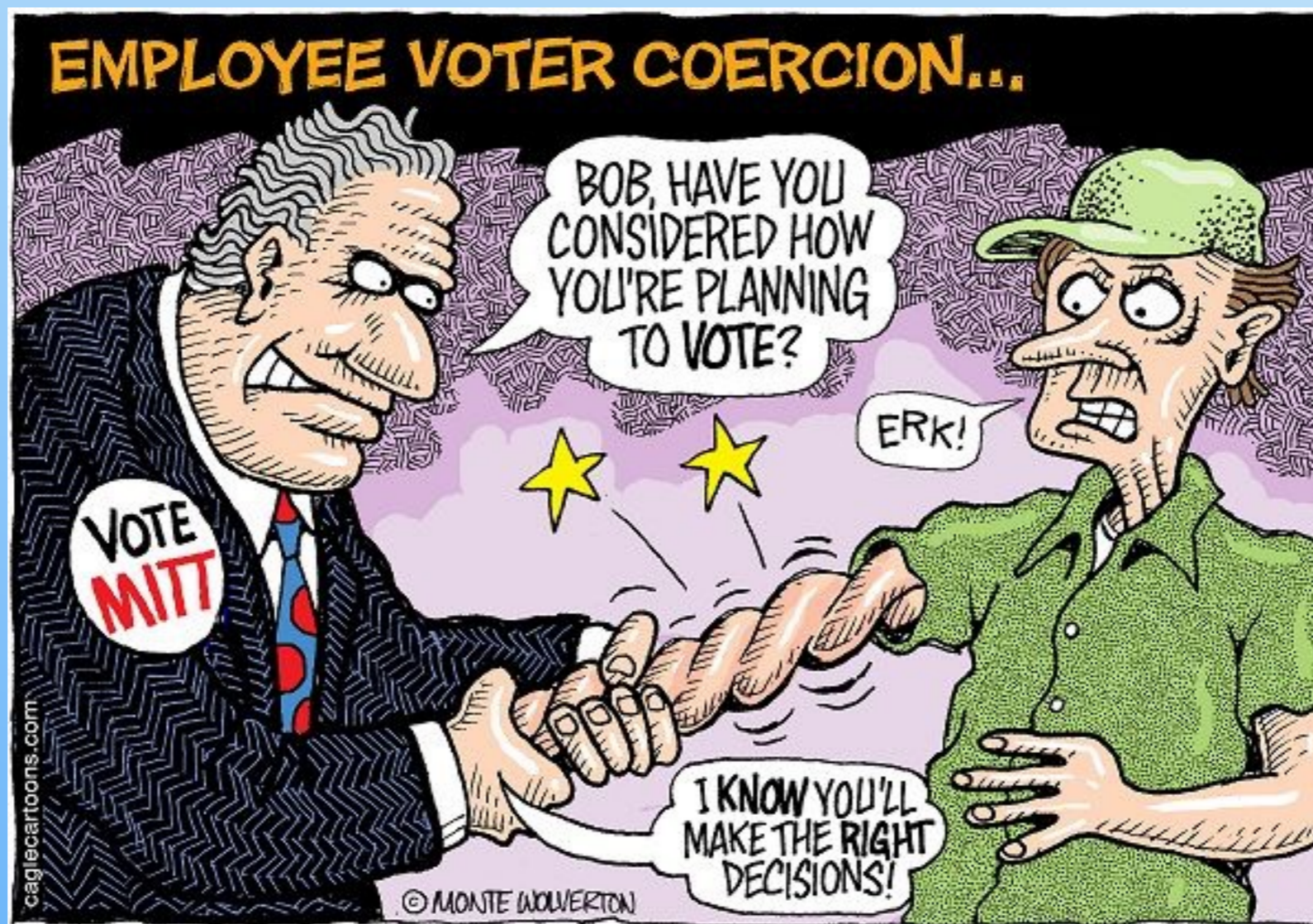
What requirements do you think a voting system should meet?



OWNED BY 'YOU'??

E-voting system requirements

What requirements do you think a voting system should meet?



Coercion-resistant ??

E-voting system - let's look inside at the code

VotingMachine1
by JP Paul Gibson

16 scripts
3 sprites

[See inside](#)

DRAFT

Instructions

This is a simple example of an e-voting machine.

It is used in my teaching as an introduction to computer science and software engineering.

It supports discussion on :

- e-voting system requirements
- ..
- ..
- ..
- ..
- ..

Notes and Credits

work-in-progress

© Shared: 15 Sep 2016 Modified: 15 Sep 2016

Electronic Voting Terminal

To open the voting process requires special authorisation

What's your authorisation to start the election



The image displays a Scratch project titled "VotingMachine1" by JP Paul Gibson. The interface on the left shows a stage with a backdrop titled "Electronic Voting Terminal" and a text box asking for authorization. Below the stage are three button sprites labeled "YES", "NO", and "CONFIRM". The right side of the image shows the Scratch script editor with three scripts:

- Script 1 (Main):** Starts when the green flag is clicked. It sets variables for "TimeLeftToVote" (0), "VoteTimePeriodInSeconds" (60), and "Administration_Password" (password). It switches the backdrop to "Voting-Teminal-StartUp" and enters a repeat loop until the user's answer matches the password. It asks for the number of voters, sets "NumberOfAuthorizedVoters" to the answer, and sets "TimeLeftToVote" to the vote period. It then enters a "forever" loop where it switches the backdrop to "Voting-Teminal-ID", checks if the answer is within the authorized range and not already voted, asks for an authorization code, and if correct, adds the voter to the list and switches the backdrop to "Voting-Teminal-Question".
- Script 2 (Timer):** Triggered by a broadcast message "start_timer", it repeats until "TimeLeftToVote" is less than 1, decreasing it by 1 each second.
- Script 3 (Result):** Triggered by a green flag click, it hides all variables and switches the backdrop to "Voting-Teminal-Result", then shows the "count_no" and "count_yes" variables.

Have any of you ever programmed in Scratch?

Key concepts : backdrop , sprites, scripts

Should e-voting system code be open/visible to the public?



```
when clicked
  set TimeLeftToVote to 0
  set VoteTimePeriodInSeconds to 60
  set Administration_Password to password
  switch backdrop to Voting-Teminal-StartUp
  repeat until answer = Administration_Password
    ask What's your authorisation to start the election and wait
  ask How many voters are on the electoral list? and wait
  set NumberOfAuthorizedVoters to answer
  set TimeLeftToVote to VoteTimePeriodInSeconds
  set count_no to 0
  set count_yes to 0
  delete all of VotersWhoHaveVoted
  show variable TimeLeftToVote
  broadcast start_timer
```

Here is the code for election set-up


```
when clicked clicked
hide
go to x: -20 y: -20
stop this script

when this sprite clicked
next costume
if costume # = 2 then
broadcast yes_on
stop this script

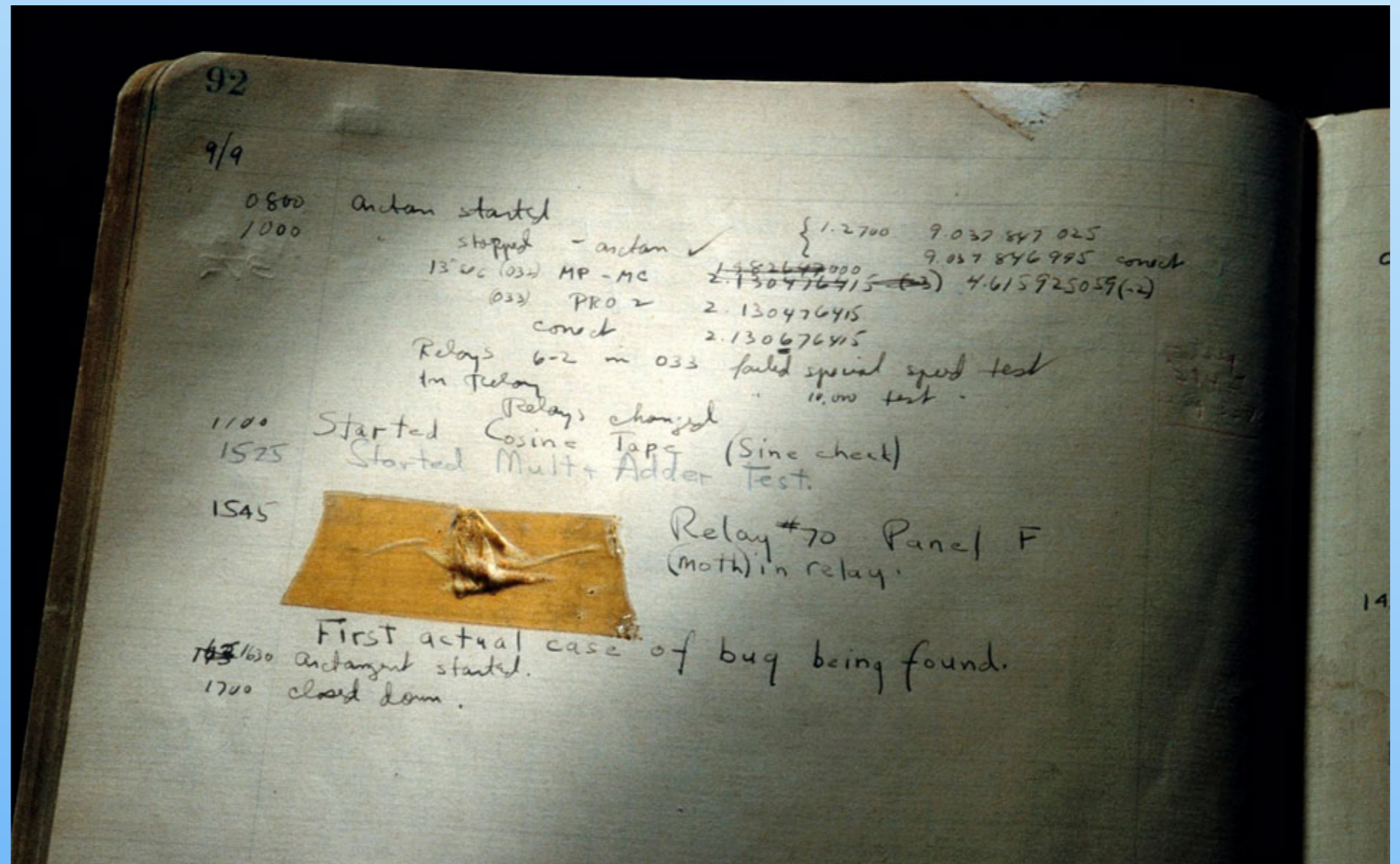
when I receive AskQuestion
switch costume to Button-No-Off
show
stop this script

when I receive no_on
switch costume to Button-Yes-Off
stop this script

when I receive confirmed
if costume # = 2 then
change count_yes by 1
switch costume to Button-Yes-Off
hide
stop this script
```

**TO DO:
can you
understand
the button
(YES) code?**

Much of the code in e-voting systems has been found to contain 'bugs'



https://en.wikipedia.org/wiki/Grace_Hopper



Much of the code in e-voting systems has been found to contain 'bugs'

When these are found, it may be :

- 1. *acceptable* human error**
- 2. incompetency**
- 3. fraud/corruption**

QUESTION: How would you *judge* between these?

QUESTION: Who is responsible for making sure that the e-voting systems are *bug-free*?

Scratch Machines On-Line:  machine 1,  machine 2,  machine 3,  machine 4

TO DO: Examine the 3 other voting machines, and find the ‘bugs’ in each of them. They make minor modifications to the first machine that we have tested/analysed. They are based on bugs that have been found in real/deployed voting systems.

QUESTION:

for each machine bug do you judge it most likely to be caused by:

- 1. *acceptable* human error, or**
- 2. *incompetency*, or**
- 3. *fraud/corruption* , or ?**