

Какво ми се иска да знаеха програмистите  
(а явно не го знаят)

Васил Колев  
vasil@ludost.net

... че имам лопата и си я държа под ръка...

# Защо водят тази лекция

- Аз трябва да накарам софтуерът да работи
- Често трябва да го debug-вам
- Редовно трябва да чета код
  - Основно за да разбере как и защо (не) работи
- Трябва да обяснявам къде е проблема, за да не се повтаря
  - Рядко помага...

# Типове проблеми

- Теоретични
- Проблеми от неинформираност
- За останалите нямам време

# Теоретични

- Copy-paste
- Умно писане
- Валидация

# Copy-Paste

- ... е метод за разпространение на грешката
- Има неща като for, функции, макроси и други неща, които го избягват
- Повтарянето на код показва или много неприятна ситуация, или неспособност на програмиста за абстрактно мислене

# Умното писане

- Debug-ването е няколко пъти по-трудно от програмирането
- Колкото по-хитро пишете, толкова по-невъзможно става дебъгването
- Много често трябва да го дебъгвам аз.
  - Аз съм тъп, и за много умен код основното ми решение е да кажа да се пренапише.

# Валидация

- Крайни автомати, формално дефинирани протоколи, defensive programming
  - Неща, които хората са чували, може би.
- Колко човека им хрумва да проверяват какво са прочели от базата данни?
  - „Само аз пиша там“ не е валидно оправдание
- Всички допускат грешки, трябва да има няколко различни места/начина за проверка



# Генерални проблеми

- Нищо не е reliable
- Distributed computing fallacies
- Version control проблеми

# Нищо не е reliable

- „I don't believe in miracles. I rely on them.“
  - Житейска философия на много програмисти
- Всичко може да върне грешка
- Failfast
- Транзакционна логика
- Идемпотентност
- ACID

```
#define BUFSZ 8192
int fd0, fd1;
size_t len;
char buff[BUFSZ];

fd0 = open(file1, O_RDONLY);
fd1 = open(file2, O_RDWR);

while ( (len = read(fd0, buff, BUFSZ) !=0 ) ) {
    write(fd1, buff, len);
}

close(fd0);
close(fd1);
```

```
#define BUFSZ 8192

#define DIE(...)

int fd0, fd1;
size_t len;
char buff[BUFSZ];

if ( ( fd0 = open(file1, O_RDONLY) ) < 0 ) DIE("open fd0");
if ( ( fd1 = open(file1, O_RDONLY) ) < 0 ) DIE("open fd1");

while ( ( len = read(fd0, buff, BUFSZ) > 0 ) ) {
    if ( write(fd1, buff, len) < len) DIE("writing");
}

if ( len < 0 ) DIE("reading");

close(fd0);
close(fd1);
```

# Distributed computing fallacies

- Продължение на темата „всички се чупи“
- Мрежата не е с нулева латентност
- Нито с безкраен bandwidth
- Не е reliable
- Не е сигурна
- И най вероятно ви мрази.
- [https://en.wikipedia.org/wiki/Fallacies\\_of\\_Distributed\\_Computing](https://en.wikipedia.org/wiki/Fallacies_of_Distributed_Computing)

# Version control

- Защо не се re-add-ват файлове, които вече ги има
  - И защо е толкова важно history-то
- Проблемът с конфигурацията във VCS-а
  - Решение с detect на средата

# Въпроси?

(не, не може да ме застреляте и проблемите  
да изчезнат :) )

Благодаря :)