



# SE302 – ПХ БҮТЭЭЛТ

## Лекц 9

# Bag of Testing Tricks

- Эквивалент хуваалт
  - Боломжит өгөгдөл оруулах нь тест тохиолдолын томоохон хэсэг
  - Хоёр тестийн тохиолдолд ижилхэн алдаа гарвал
    - Тэндний нэг нь л хэрэгтэй
    - Ийм байдлаар тестийн тохиолдлыг бууруулах аргачлалыг эквивалент хуваалт гэнэ

## Example of Computing the Number of Cases Needed for Basis Testing of a Java Program

```
► 1 // Compute Net Pay
  2 totalwithholdings = 0;
  3
► 4 for ( id = 0; id < numEmployees; id++ ) {
  5
  6     // compute social security withholding, if below the maximum
► 7     if ( m_employee[ id ].governmentRetirementwithheld < MAX_GOVT_RETIREMENT ) {
  8         governmentRetirement = ComputeGovernmentRetirement( m_employee[ id ] );
  9     }
```

# Bag of Testing Tricks

- ▶ Хоёр хэсэгт хувааж авч үзнэ
  - ▶ `m_employee[ ID ].governmentRetirementWithheld` нь бага `MAX_GOVT_RETIREMENT`
  - ▶ `m_employee[ ID ].governmentRetirementWithheld` нь их буюу тэнцүү `MAX_GOVT_RETIREMENT`
- ▶ Хоёр болон түүнээс дээш боломжит утгыг шалгаж үзэх хэрэгтэй

# Bag of Testing Tricks

- Алдааг таах / Error Guessing
  - Сайн програмистууд албан бус тестийн арга техник ашигладаг тал бий
  - Сэтгэн олох аргаар өөрийн кодны алдааг илрүүлэх
  - Програмын алдаа байж магадгүй гэсэн таамаглал дээр тулгуурлаж тестийн тохиолдол үүсгэх
  - Өөрийн зөн совин болон өнгөрсөн туршлага
    - Түгээмэл алдааны жагсаалт
    - Өмнөх өөрийн гаргадаг алдааны төрөл болон алдааны бичлэгүүд

# Bag of Testing Tricks

- Хил хязгаарыг шинжлэх / Boundary Analysis
  - Тестэд үр өгөөжтэй зүйлсийн нэг нь хил хязгаарын нөхцөл
    - `num - 1` гэж бодох үед `num` байх
    - `>` гэж бодох үед `>=` байх
  - Хил хязгаарын нөхцөлд тестийн тохиолдол үүсгэх
    - Мах-аас бага байх гэдэг нөхцөлд тестийн тохиолдол үүсгэхэд
      - Дараах 3 тестийн тохиолдол үүсгэнэ



# Bag of Testing Tricks

- If (`m_employee[ ID ].governmentRetirementWithheld < MAX_GOVT_RETIREMENT`)
  - Case 1:
    - `MAX_GOVT_RETIREMENT - 1` хүртэл
      - Нөхцөл үнэн үед
      - Өмнө нь үүссэн тестийн тохиолдол
  - Case 3:
    - `MAX_GOVT_RETIREMENT + 1` дээш
      - Нөхцөл худал үед
      - Өмнө нь үүссэн тестийн тохиолдол
  - Case 10:
    - `m_employee [ ID ].governmentRetirementWithheld = MAX_GOVT_RETIREMENT`
      - Шинэ тестийн тохиолдол. (boundary analysis)



# Bag of Testing Tricks

- Нийлмэл хил хязгаар / Compound Boundary
  - Boundary analysis-д байж болох хамгийн их, бага утгыг зааж болно
    - Нийт цалин, Тэтгэвэрт гаралт, Хадгалуулсан мөнгө
  - Хоёр тоог үржихэд
    - Том эерэг тоонууд?
    - Том сөрөг тоонууд?
    - 0?
    - Хэт урт тэмдэгт мөр дамжуулах?
  - Олон гишүүнтэй том бүлэгийн гишүүн бүр өндөр цалинтай, Програмистуудын бүлэг 250,000\$
    - 1000 гишүүнтэй үед
    - Гишүүн бүрийн цалин 0.00\$ үед



# Bag of Testing Tricks

- Муу өгөгдөл
  - Алдааг тааварлахаас гадна, дараах муу өгөгдлүүдийг ашиглаж болно
    - Хэт бага өгөгдөл ( эсвэл өгөгдөлгүй)
    - Хэт их өгөгдөл
    - Буруу төрлийн өгөгдөл (буруу өгөгдөл)
    - Буруу хэлбэрийн өгөгдөл
    - Ачаалагдаагүй өгөгдөл
  - 100,000,000 ажилтантай байх
  - Сөрөг цалин
  - Сөрөг тооны ажилтан



# Bag of Testing Tricks

## ■ Сайн өгөгдөл

- Програмаас алдаа хайхад
  - Төлөвлөгдсөн утгууд нь хялбар болгож өгдөг
- Үндсэн тестийн аргад ихэвчлэн төлөвлөгдсөн утгууд нь тодорхойлогдсон байдаг
- Сайн өгөгдлүүд нь шалгалтыг үр дүнтэй болгодог
  - Төлөвлөгдсөн утгууд, дунд нь
  - Хамгийн бага хэвийн тохиргоо
  - Хамгийн их хэвийн тохиргоо
  - Хуучин өгөгдөлтэй зохицох

# Typical Errors

- Аль класс ихэнх алдааг агуулдаг вэ?
  - Алдааг өөрийн кодонд жигд тархсан байна гэж боддог
    - 1000 мөртэй кодонд 10 алдаа илэрсэн
    - 100 мөр бүрт нэг алдаа байна гэж үзэх нь буруу
  - IBM-ийн IMS-н чанарыг сайжруулахаар төвлөрсөн
    - 425 классын 31 нь алдаатай байсан
    - Нэг жил хүрэхгүй хугацаанд хөгжүүлэлт хийсэн
      - Хэрэглэгчдээс ирсэн алдааг 10-с 1 болтол бууруулсан
    - Техникийн үйлчилгээний зардал нь 45% болтол буурсан
    - Хэрэглэгчийн сэтгэл ханамж
      - Хүлээж авах боломжгүй -> сайн үнэлгээ авсан

# Typical Errors

- Ихэнх алдаа нь цөөн хэдэн сул функцуудэд цуглардаг
- Алдаа болон кодын ерөнхий хамаарал
  - Алдааны 50% нь төслийн 20%-ийн класс эсвэл функцэд агуулагддаг
    - Endres 1975, Gremillion 1984, Boehm 1987b, Shull et al 2002
  - Алдааны 50% нь төслийн классын 5%-д агуулагддаг
    - Jones 2000



# Typical Errors

25.18%	Structural
22.44%	Data
16.19%	Functionality as implemented
9.88%	Construction
8.98%	Integration
8.12%	Functional requirements
2.76%	Test definition or execution
1.74%	System, software architecture
4.71%	Unspecified

# Test-Support Tools

- Мэдээлэл бичих / лог бүртгэл
  - Зарим tools програмыг удирдаж, алдааны төлөвүүдийг цуглуулдаг
  - Онгоцны хар хайрцаг шиг
- Сайн лог нь Програм гарсанаас хойш алдааг засах сервисэд тусалдаг
- Өөрийн алдааны тухай дэлгэрэнгүй мэдээлэлдэг лог бичигчийг бүтээж болно
- Хөгжүүлэлтийн хувилбарт агуулагдана
  - Бүтээгдхүүнд орхиж хөрвүүлэлт хийнэ
  - Өөрөө тайрах, ухаалаг байрлуулах асуудлыг шийдвэл бүтээгдхүүнд агуулагдаж болно

# Test-Support Tools

## ▶ Debugger

- ▶ Кодыг мөр мөрөөр шинжлэнэ
  - ▶ Хувьсагчдын утгуудыг хянаана
  - ▶ Тухайн агшинд кодыг интерпрет хийнэ
  - ▶ Кодын хэсгүүдийг харж, хэрхэн ажиллаж байгааг үзнэ
- ▶ Бусад програмистууд таны кодыг шалгахад туслах
  - ▶ Багаар шалгах ажилаас бага хөдөлмөр шаарддаг



# Test-Support Tools

- System Perturbers
  - Олон хүмүүст тохиолддог
    - 100-с 99 удаа ажилладаг
    - Гэвч нэг удаа алдаа гардаг
      - Ижилхэн өгөгдөл дээр
  - Ийм алдаа ихэнхдээ хувьсагч `init` хийх алдаа байдаг
  - `Init` хийгээгүй хувьсагч 100-ийн 99-д нь 0 гэсэн утгатай байдаг

# Test-Support Tools

- Test-support tools -ийн боломжууд
  - Memory filling
    - Init хийгээгүй хувьсагч байхгүй гэдэгт итгэлтэй байх хэрэгтэй
    - Зарим tools санах ойг тохиолдлын утгаар дүүргэдэг
    - Init хийгдээгүй утгууд 0 гэж тохируулагдахгүй
    - Санах ойд онцгой утга байж болно
      - 0xCC - машины хэлний код
      - Дебагын Breakpoint тавьж алдааг илрүүлэх
  - Memory shaking
    - Multitasking системд, зарим tools санах ойг дахин зохион байгуулах
    - Absolute утгаар биш relative утгаар ханддаг байх

# Test-Support Tools

- Test-support tools -ийн боломжууд
  - Selective memory failing
    - Санах ой удирдах simulator
      - Санах ой бага үед програм санах ойгоос халих
      - Санах ой хүсэх алдаа гарах
    - Хэцүү бэрх үед хэрхэн давах тест
      - Санах ойг динамикаар хувиарлах
  - Memory-access checking (bounds checking)
    - Заагч нь өөрсдөд дээрээ ажилладаг байх
    - Ийм төрлийн хэрэгсэл нь
      - Init хийгдээгүй хувьсагч, дээрх төрлийн бус заагчыг илрүүлдэг

# Test-Support Tools

- Improving your testing
  - Тестийг сайжруулах алхам нь бусад процессыг сайжруулах алхамуудтай төстэй
    - Нэг үр ашигтай зүйл нь төслийн эхэнд тест хийх төлөвлөгөө гаргах
    - Тестийг дизайн болон кодчилолтой ижил түвшинд авч үзэх
  - Тестийг төлөвлөх нь тестийн процессийг давтаж хийхийн нэг элемент
    - Хэрэв давтаж чадахгүй бол сайжруулж чадахгүй

# Test-Support Tools

- Retesting (Regression testing)
  - Бүтээгдхүүнийг тестлээд алдаа илрүүлээгүй гэж үзье
    - Програмын нэг хэсэгт өөрчлөлт орсон
    - Өөрчлөлтөөс өмнө энэ нь тестийг давсан байсан
    - Өөрчлөлтөөр ямар нэгэн алдаа илрээгүй
  - Өөрчлөлтийн дараа системчилсэн дахин тест хийнэ
    - Өөр тест ажилуулвал, өөрчлөлтөөр шинэ алдаа гараагүй гэдэгт итгэх боломжгүй
  - Regression тест нь тест хийх бүрд ижилхэн байх ёстой
    - Заримдаа шинээр нэмэгдэж болно
    - Хуучин тестийн тохиолдол хадгалагдах

# Test-Support Tools

- Automated Testing
  - Практикт regression тестийг автоматжуулдаг
    - Хүн олон удаа давтахад алдаа гаргадаг
    - Алдааг харахад хялбар болгодог нь regression тестийн зорилго
  - Давуу талууд
    - Буруу ажиллах нь бага
    - Нэг л удаа автоматжуул, цаашид ажил хөнгөрнө
    - Автомат тест нь алдааг аль болох эрт илрүүлэх боломжыг нэмэгдүүлнэ
    - Хувирамтгай (volatile) технологийн орчинд онцгой хэрэгтэй



# Test-Support Tools

- Keeping test records
  - Тестийн процесс давтагдах ёстой
  - Төсөл сайжирсан эсвэл муудсан гэдгийг хэмжих хэрэгтэй
    - Алдааны тухай админы тайлбар(огноо, хэн тайлагнасан, гарчиг эсвэл тайлбар, build-ийн дугаар, зассан огноо)
    - Алдааны дэлгэрэнгүй тайлбар
    - Асуудлыг дахин авах алхам
    - Асуудлыг шийдэх болон тойрон гарах арга зам
    - Хамаарлуудыг илрүүлэх
    - Асуудлын аюултай байдал
      - Fatal, cosmetic ...
    - Алдааг илрүүлсэн гарал
      - Шаардлага, дизайн, код, тест



# Test-Support Tools

- Keeping test records
  - Класс, функцууд засагдсан
  - Хэдэн мөр код алдаанд нөлөөлсөн
  - Алдааг илрүүлсэн хугацаа
  - Алдааг зассан хугацаа