



Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

Herramientas para detectar bugs

Seminario de Lenguajes opción C

Facultad de Informática
Universidad Nacional de la Plata

2016



Indice

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

① Introducción

② Análisis estático

Introducción

cppcheck

splint

③ Análisis en tiempo de ejecución

Introducción

valgrind

gdb



Introducción

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Análisis estático
- Análisis en tiempo de ejecución
- Muy útiles, pero no son perfectos:
 - Falsos positivos
 - No detectan todo



Introducción

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Análisis estático
- Análisis en tiempo de ejecución
- Muy útiles, pero no son perfectos:
 - Falsos positivos
 - No detectan todo



Introducción

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Análisis estático
- Análisis en tiempo de ejecución
- Muy útiles, pero no son perfectos:
 - Falsos positivos
 - No detectan todo



Introducción

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Análisis estático
- Análisis en tiempo de ejecución
- Muy útiles, pero no son perfectos:
 - Falsos positivos
 - No detectan todo



Introducción

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Análisis estático
- Análisis en tiempo de ejecución
- Muy útiles, pero no son perfectos:
 - Falsos positivos
 - No detectan todo



Análisis estático

Introducción

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Se ejecutan sobre el código fuente
- Complementan las verificaciones del compilador



Análisis estático

cppcheck

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

Detecta entre otros¹:

- Desreferencias de punteros NULL
- Memory leaks
- Divisiones por cero
- Condiciones sospechosas (asignación + comparación)
- Código inalcanzable
- ...

Uso:

```
cppcheck dir # Recursivamente en dir  
cppcheck --enable=all archivo.c # Más pruebas
```

¹<http://sourceforge.net/apps/mediawiki/cppcheck/index.php?title=ListOfChecks>



Análisis estático

cppcheck

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

Detecta entre otros¹:

- Desreferencias de punteros NULL
- Memory leaks
- Divisiones por cero
- Condiciones sospechosas (asignación + comparación)
- Código inalcanzable
- ...

Uso:

```
cppcheck dir # Recursivamente en dir  
cppcheck --enable=all archivo.c # Más pruebas
```

¹<http://sourceforge.net/apps/mediawiki/cppcheck/index.php?title=ListOfChecks>



Análisis estático

splint

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Más estricto que cppcheck
- Detecta más errores
- La salida es más engorrosa
- Falla más seguido que cppcheck parsear

Uso:

```
splint archivo.c
```



Análisis estático

splint

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Más estricto que cppcheck
- Detecta más errores
- La salida es más engorrosa
- Falla más seguido que cppcheck parsear

Uso:

```
splint archivo.c
```



Análisis en tiempo de ejecución

Introducción

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Se realizan sobre el programa compilado
- Compilar con la opción `-g`
- Detectan errores imposibles de detectar de forma estática
- No verifican el código que no se ejecuta



Análisis en tiempo de ejecución

Introducción

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Se realizan sobre el programa compilado
- Compilar con la opción `-g`
- Detectan errores imposibles de detectar de forma estática
- No verifican el código que no se ejecuta



Análisis en tiempo de ejecución

Introducción

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Se realizan sobre el programa compilado
- Compilar con la opción `-g`
- Detectan errores imposibles de detectar de forma estática
- No verifican el código que no se ejecuta



Análisis en tiempo de ejecución

Introducción

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Se realizan sobre el programa compilado
- Compilar con la opción `-g`
- Detectan errores imposibles de detectar de forma estática
- No verifican el código que no se ejecuta



Análisis en tiempo de ejecución

valgrind

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Accesos a memoria no alocada
- Memory leaks
- Estos errores son los más comunes

Uso:

```
gcc -Wall -g -o programa prog1.c prog2.c  
valgrind ./programa  
valgrind --leak-check=full ./programa
```



Análisis en tiempo de ejecución

valgrind

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Accesos a memoria no alocada
- Memory leaks
- Estos errores son los más comunes

Uso:

```
gcc -Wall -g -o programa prog1.c prog2.c  
valgrind ./programa  
valgrind --leak-check=full ./programa
```



Análisis en tiempo de ejecución

valgrind

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Accesos a memoria no alocada
- Memory leaks
- Estos errores son los más comunes

Uso:

```
gcc -Wall -g -o programa prog1.c prog2.c  
valgrind ./programa  
valgrind --leak-check=full ./programa
```



Análisis en tiempo de ejecución

valgrind

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Accesos a memoria no alocada
- Memory leaks
- Estos errores son los más comunes

Uso:

```
gcc -Wall -g -o programa prog1.c prog2.c  
valgrind ./programa  
valgrind --leak-check=full ./programa
```



Análisis en tiempo de ejecución

valgrind

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Accesos a memoria no alocada
- Memory leaks
- Estos errores son los más comunes

Uso:

```
gcc -Wall -g -o programa prog1.c prog2.c  
valgrind ./programa  
valgrind --leak-check=full ./programa
```



Análisis en tiempo de ejecución

gdb

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Debugger de gcc
 - Varios frontends: ddd, cgdb, kdbg, netbeans, eclipse, etc...
 - Breakpoints
 - Watchpoints
 - Comandos
 - run
 - break
 - display
 - continue
 - next
 - nexti
 - until
 - delete
 - info [break/watchpoints]
 - list



Análisis en tiempo de ejecución

gdb

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Debugger de gcc
- Varios frontends: ddd, cgdb, kdbg, netbeans, eclipse, etc...

- Breakpoints
- Watchpoints
- Comandos

`run`

`break`

`display`

`continue`

`next`

`nexti`

`until`

`delete`

`info [break/watchpoints]`

`list`



Análisis en tiempo de ejecución

gdb

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Debugger de gcc
- Varios frontends: ddd, cgdb, kdbg, netbeans, eclipse, etc...
- Breakpoints
- Watchpoints
- Comandos
 - `run`
 - `break`
 - `display`
 - `continue`
 - `next`
 - `nexti`
 - `until`
 - `delete`
 - `info [break/watchpoints]`
 - `list`



Análisis en tiempo de ejecución

gdb

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Debugger de gcc
- Varios frontends: ddd, cgdb, kdbg, netbeans, eclipse, etc...
- Breakpoints
- Watchpoints
- Comandos
 - `run`
 - `break`
 - `display`
 - `continue`
 - `next`
 - `nexti`
 - `until`
 - `delete`
 - `info [break/watchpoints]`
 - `list`



Análisis en tiempo de ejecución

gdb

Herramientas
para detectar
bugs

Seminario de
Lenguajes
opción C

Introducción

Análisis
estático

Introducción
cppcheck
splint

Análisis en
tiempo de
ejecución

Introducción
valgrind
gdb

- Debugger de gcc
- Varios frontends: ddd, cgdb, kdbg, netbeans, eclipse, etc...
- Breakpoints
- Watchpoints
- Comandos

`run`

`break`

`display`

`continue`

`next`

`nexti`

`until`

`delete`

`info [break/watchpoints]`

`list`