Valgrind tutorial

LABC

Get & install

- *homepage: http://www.valgrind.org/*
- <u>install:</u>
  - extract:
    - bzip2 -d valgrind-XYZ.tar.bz2
    - tar -xf valgrind-XYZ.tar
  - which will create a directory called valgrind-XYZ; change into that directory and run:
    - ./configure
    - make
    - make install
- Many linux dists. come with prepared package, google 'dist-name valgrind'

Memory leaks (1)

```
//file: exp1.c
#include <stdlib.h>
int main()
{
    char *x = malloc(100);
    return 0;
}
```

compile:

```
$>   gcc -Wall -g exp1.c -o exp1
```

Memory leaks (1) - code

```
//file: exp1.c
#include <stdlib.h>
int main()
{
    char *x = malloc(100);
    return 0;
}
```

compile:

```
$>   gcc -Wall -g exp1.c -o exp1
// -g for debug inf.
```

Memory leaks (1) – run valgrind

run:

```
$>  valgrind exp1
ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 11 from 1)
==26064== malloc/free: in use at exit: 100 bytes in 1 blocks.
==26064== malloc/free: 1 allocs, 0 frees, 100 bytes allocated.
==26064== For counts of detected errors, rerun with: -v
==26064== searching for pointers to 1 not-freed blocks.==26064== checked 52,096
    bytes.
==26064==
==26064== LEAK SUMMARY:
==26064==    definitely lost: 100 bytes in 1 blocks.
==26064==    possibly lost: 0 bytes in 0 blocks.
==26064==    still reachable: 0 bytes in 0 blocks.
==26064==    suppressed: 0 bytes in 0 blocks.
==26064==    Use --leak-check=full to see details of leaked memory.
```

Memory leaks (1) – run valgrind

run:

```
$>  valgrind exp1
ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 11 from 1)
==26064== malloc/free: in use at exit: 100 bytes in 1 blocks.
==26064== malloc/free: 1 allocs, 0 frees, 100 bytes allocated.
==26064== For counts of detected errors, rerun with: -v
==26064== searching for pointers to 1 not-freed blocks.==26064== checked 52,096
    bytes.
==26064==
==26064== LEAK SUMMARY:
==26064==    definitely lost: 100 bytes in 1 blocks.
==26064==    possibly lost: 0 bytes in 0 blocks.
==26064==    still reachable: 0 bytes in 0 blocks.
==26064==    suppressed: 0 bytes in 0 blocks.
==26064==    Use --leak-check=full to see details of leaked memory.
```

Memory leaks (1) – run valgrind

run:

```
$>  valgrind exp1
ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 11 from 1)
==26064== malloc/free: in use at exit: 100 bytes in 1 blocks.
==26064== malloc/free: 1 allocs, 0 frees, 100 bytes allocated.
==26064== For counts of detected errors, rerun with: -v
==26064== searching for pointers to 1 not-freed blocks.==26064== checked 52,096
    bytes.
==26064==
==26064== LEAK SUMMARY:
==26064==    definitely lost: 100 bytes in 1 blocks.
==26064==    possibly lost: 0 bytes in 0 blocks.
==26064==    still reachable: 0 bytes in 0 blocks.
==26064==    suppressed: 0 bytes in 0 blocks.
==26064==    Use --leak-check=full to see details of leaked memory.
```

Memory leaks (1) – run valgrind

run:

```
$>  valgrind --leak-check=full  exp1
```

…

```
==32353== 100 bytes in 1 blocks are definitely lost in loss record 1 of 1
==32353==    at 0x4004405: malloc (vg_replace_malloc.c:149)
==32353==    by 0x8048370: main (exp1.c:4)
```

…

Unallocated memory (2)

```
//file: exp2.c
#include <stdlib.h>
int main()
{
  char *x = malloc(10);
  x[10] = 'a';
  return 0;
}
```

compile:

```
$>   gcc -Wall -g exp1.c -o exp2
```

Unallocated mem (2) – run valgrind

run:

```
$>  valgrind exp2

==26190== Invalid write of size 1
==26190== at 0x804837A: main (exp2.c:6)
==26190== Address 0x413C032 is 0 bytes after a block of size 10 alloc'd
==26190== at 0x401D38B: malloc (vg_replace_malloc.c:149)
==26190== by 0x8048370: main (exp2.c:5)
==26190==
==26190== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 11
   from 1)
```

Unallocated mem (2) – run valgrind invalid write

run:

```
$>  valgrind exp2

==26190== Invalid write of size 1
==26190== at 0x804837A: main (exp2.c:6)
==26190== Address 0x413C032 is 0 bytes after a block of size 10 alloc'd
==26190== at 0x401D38B: malloc (vg_replace_malloc.c:149)
==26190== by 0x8048370: main (exp2.c:5)
==26190==
==26190== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 11
   from 1)
```

Unallocated mem (2) – run valgrind invalid write – where?

run:

```
$>  valgrind exp2

==26190== Invalid write of size 1
==26190== at 0x804837A: main (exp2.c:6)
==26190== Address 0x413C032 is 0 bytes after a block of size 10 alloc'd
==26190== at 0x401D38B: malloc (vg_replace_malloc.c:149)
==26190== by 0x8048370: main (exp2.c:5)
==26190==
==26190== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 11
   from 1)
```

Uninitialized values (3)

```
//file: exp3.c
#include <stdio.h>
int main()
{
  int x;
  if(x == 0)
  {
      printf("X is zero");
  }
  return 0;
}
```

Uninitialized values (3) – run valgrind

run:

```
$>  valgrind exp3

==26127==    Conditional jump or move depends on uninitialised value(s)
==26127==    at 0x8048369: main (exp3.c:6)
```

Uninitialized values (3) – run valgrind

run:

```
$>  valgrind exp3

==26127==    Conditional jump or move depends on uninitialised value(s)
==26127==    at 0x8048369: main (exp3.c:6)
```

Uninitialized values (3) – run valgrind where?

run:

```
$>  valgrind exp3

==26127==    Conditional jump or move depends on uninitialised value(s)
==26127==    at 0x8048369: main (exp3.c:6)
```

Uninitialized values (4)

```
//file: exp4.c
#include <stdio.h>
int foo(int x)
{
  if(x < 10)
  {
   printf("x is less
           than 10\n");
   }
  return 0;
}
```

```
int main()
{
  int y;
  foo(y);
  return 0;
}
```

Uninitialized values (4) – run valgrind

run:

```
$>  valgrind exp4

==26128==    Conditional jump or move depends on uninitialised value(s)
==26128==    at 0x804835E: foo (exp4.c:5)
==26128==    by 0x804838E: main (exp4.c:15)
```

Seg faults (5)

```
//file: exp5.c
int main()
{
  char x[10];
  x[11] = 'a';
  return 0;
}
```

Seg faults (5) – run valgrind

run:

$> valgrind exp5

==26131== Invalid read of size 4

==26131== at 0x8048346: main (exp5.c:6)

==26131== Address 0xBE9561BC is not stack'd, malloc'd or (recently) free'd

Seg faults (5) – run valgrind

run:

$> valgrind exp5

```
==26131==
==26131== Process terminating with default action of signal 11 (SIGSEGV)
==26131== Access not within mapped region at address 0xBE9561BC
==26131== at 0x8048346: main (exp5.c:6)
==26131==
==26131== Process terminating with default action of signal 11 (SIGSEGV)
==26131== Access not within mapped region at address 0xBE9561B8
==26131== at 0x40191E0: _vgnU_freeres (vg_preloaded.c:56)
==26131==
==26131== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 11
   from 1)
```

What valgrind can’t do (6)?

```
//file: exp6.c
#include <stdio.h>
int main()
{
  unsigned int a = 30;
  unsigned int b = 20;
  unsigned int dif = b - a;
  printf ("%ud\n", dif);
  return 0;
}
```

What valgrind can’t do (6)?

run:

$> valgrind exp6

==26132==

4294967286d

==26132==

==26132== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 11 from 1)

Invalid free (7)

```
//file: exp7.c
#include <stdlib.h>
int main()
{
  int *arr = (int*) malloc (10*sizeof
  (int));
  arr = arr + 1;
  free (arr);
  return 0;
}
```

Invalid free (7) – run valgrind

run:

$> valgrind exp7

```
==26133== Invalid free() / delete / delete[]
==26133== at 0x401CFA5: free (vg_replace_malloc.c:233)
==26133== by 0x80483C2: main (exp7.c:9)
==26133== Address 0x413C02C is 4 bytes inside a block of size 40 alloc'd
==26133== at 0x401D38B: malloc (vg_replace_malloc.c:149)
==26133== by 0x80483B0: main (exp7.c:5)
+=26133==
==26133== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 11
   from 1)
==26133== malloc/free: in use at exit: 40 bytes in 1 blocks.
==26133== malloc/free: 1 allocs, 1 frees, 40 bytes allocated.
```

Invalid free (7) – run valgrind

run:

$> valgrind exp7

```
==26133== Invalid free() / delete / delete[]
==26133== at 0x401CFA5: free (vg_replace_malloc.c:233)
==26133== by 0x80483C2: main (exp7.c:9)
==26133== Address 0x413C02C is 4 bytes inside a block of size 40 alloc'd
==26133== at 0x401D38B: malloc (vg_replace_malloc.c:149)
==26133== by 0x80483B0: main (exp7.c:5)
+=26133==
==26133== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 11
   from 1)
==26133== malloc/free: in use at exit: 40 bytes in 1 blocks.
==26133== malloc/free: 1 allocs, 1 frees, 40 bytes allocated.
```

When all looks o.k. (8)

```
//file: exp8.c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main( int argc, char ** argv )
{
  char * snGreeting = malloc( sizeof(char) * 1024 );
  strcpy( snGreeting, "hello" );
  free(snGreeting);
  printf( "%s Sir/Madam\n", snGreeting );
  return 0;
}
```

When all looks o.k. (8)

```
//file: exp8.c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int main( int argc, char ** argv )
{
  char * greeting = malloc( sizeof(char) * 1024 );
  strcpy( greeting, "hello" );
  free(greeting);
  printf( "%s Sir/Madam\n", greeting );
  return 0;
}
```

When all looks o.k. (8) – run

run:

```
$> exp8
hello Sir/Madam
$>
```

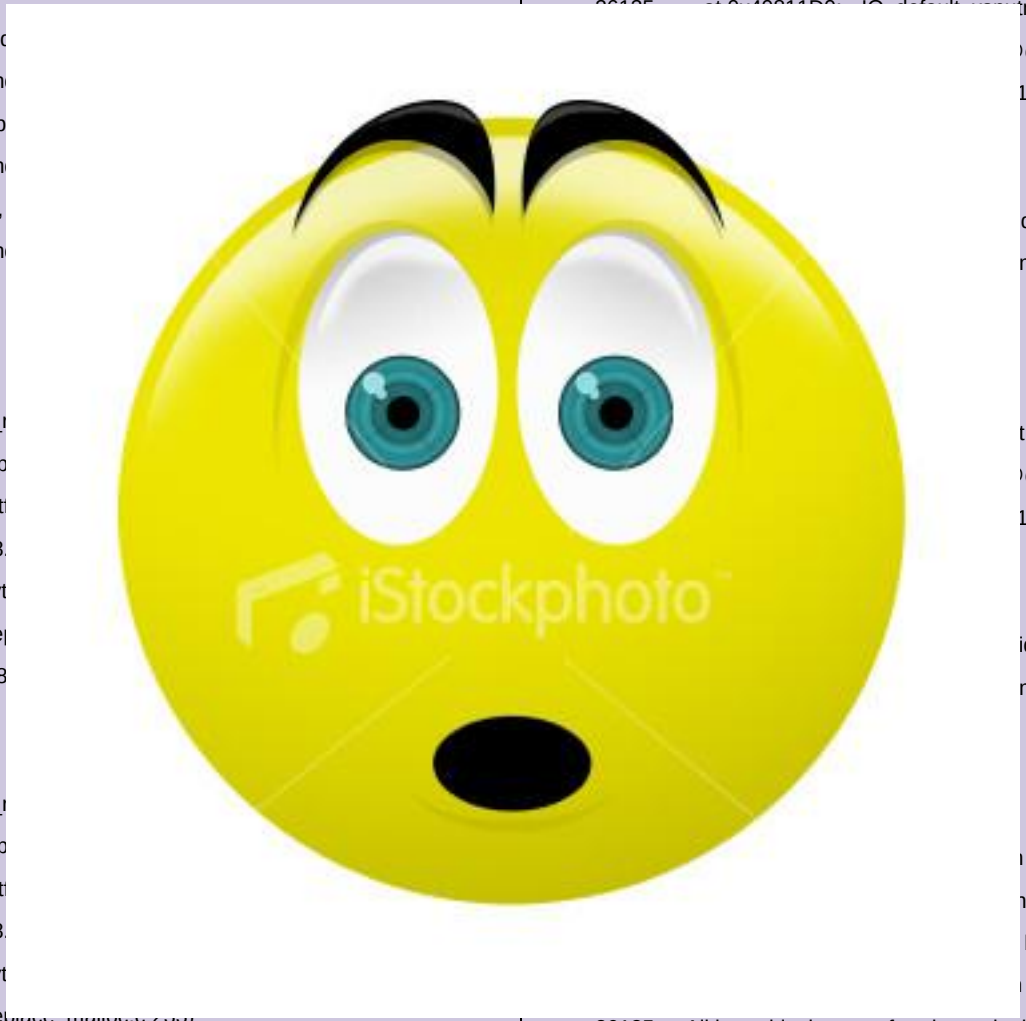
And with valgrind:

When all looks o.k. (8) – run valgrind

```
run:
$>  valgrind exp8
==26135== Memcheck, a memory error detector.
==26135== Copyright (C) 2002-2006, and GNU GPL'd, by Julian Seward et al.
==26135== Using LibVEX rev 1658, a library for dynamic binary translation.
==26135== Copyright (C) 2004-2006, and GNU GPL'd, by OpenWorks LLP.
==26135== Using valgrind-3.2.1-Debian, a dynamic binary instrumentation framework.
==26135== Copyright (C) 2000-2006, and GNU GPL'd, by Julian Seward et al.
==26135== For more details, rerun with: -v
==26135==
==26135== Invalid read of size 1
==26135==    at 0x401E208: strlen (mc_replace_strmem.c:246)
==26135==    by 0x405F0C7: vfprintf (vfprintf.c:1535)
==26135==    by 0x4064C72: printf (printf.c:34)
==26135==    by 0x8048410: main (exp8.c:10)
==26135==  Address 0x413C028 is 0 bytes inside a block of size 1,024 free'd
==26135==    at 0x401CFA5: free (vg_replace_malloc.c:233)
==26135==    by 0x80483FD: main (exp8.c:9)
==26135==
==26135== Invalid read of size 1
==26135==    at 0x401E211: strlen (mc_replace_strmem.c:246)
==26135==    by 0x405F0C7: vfprintf (vfprintf.c:1535)
==26135==    by 0x4064C72: printf (printf.c:34)
==26135==    by 0x8048410: main (exp8.c:10)
==26135==  Address 0x413C029 is 1 bytes inside a block of size 1,024 free'd
==26135==    at 0x401CFA5: free (vg_replace_malloc.c:233)
==26135==    by 0x80483FD: main (exp8.c:9)
==26135==
==26135== Invalid read of size 1
==26135==    at 0x40811D0: _IO_default_xsputn (genops.c:470)
==26135==    by 0x407F02B: _IO_file_xsputn@@GLIBC_2.1 (fileops.c:1360)
==26135==    by 0x405F071: vfprintf (vfprintf.c:1535)
==26135==    by 0x4064C72: printf (printf.c:34)
==26135==    by 0x8048410: main (exp8.c:10)
==26135==  Address 0x413C028 is 0 bytes inside a block of size 1,024 free'd
==26135==    at 0x401CFA5: free (vg_replace_malloc.c:233)
==26135==    by 0x80483FD: main (exp8.c:9)
==26135==
==26135== Invalid read of size 1
==26135==    at 0x40811DA: _IO_default_xsputn (genops.c:469)
==26135==    by 0x407F02B: _IO_file_xsputn@@GLIBC_2.1 (fileops.c:1360)
==26135==    by 0x405F071: vfprintf (vfprintf.c:1535)
==26135==    by 0x4064C72: printf (printf.c:34)
==26135==    by 0x8048410: main (exp8.c:10)
==26135==  Address 0x413C02A is 2 bytes inside a block of size 1,024 free'd
==26135==    at 0x401CFA5: free (vg_replace_malloc.c:233)
==26135==    by 0x80483FD: main (exp8.c:9)
hello Sir/Madam
==26135==
==26135== ERROR SUMMARY: 11 errors from 4 contexts (suppressed: 11 from 1)
==26135== malloc/free: in use at exit: 0 bytes in 0 blocks.
==26135== malloc/free: 1 allocs, 1 frees, 1,024 bytes allocated.
==26135== For counts of detected errors, rerun with: -v
==26135== All heap blocks were freed -- no leaks are possible.
```

When all looks o.k. (8) – run valgrind

```
run:
$>  valgrind exp8
==26135== Memcheck, a memory error c
==26135== Copyright (C) 2002-2006, an
==26135== Using LibVEX rev 1658, a lib
==26135== Copyright (C) 2004-2006, an
==26135== Using valgrind-3.2.1-Debian,
==26135== Copyright (C) 2000-2006, an
==26135== For more details, rerun with:
==26135==
==26135== Invalid read of size 1
==26135==    at 0x401E208: strlen (mc_r
==26135==    by 0x405F0C7: vfprintf (vfp
==26135==    by 0x4064C72: printf (printf
==26135==    by 0x8048410: main (exp8.
==26135==  Address 0x413C028 is 0 byt
==26135==    at 0x401CFA5: free (vg_rep
==26135==    by 0x80483FD: main (exp8
==26135==
==26135== Invalid read of size 1
==26135==    at 0x401E211: strlen (mc_r
==26135==    by 0x405F0C7: vfprintf (vfp
==26135==    by 0x4064C72: printf (printf
==26135==    by 0x8048410: main (exp8.
==26135==  Address 0x413C029 is 1 byt
==26135==    at 0x401CFA5: free (vg_replace_malloc.c:233)
==26135==    by 0x80483FD: main (exp8.c:9)
==26135==
```




```
==26135== Invalid read of size 1
                                  tn (genops.c:470)
                                  @GLIBC_2.1 (fileops.c:1360)
                                  1535)

                                  de a block of size 1,024 free'd
                                  malloc.c:233)

                                  tn (genops.c:469)
                                  @GLIBC_2.1 (fileops.c:1360)
                                  1535)

                                  ide a block of size 1,024 free'd
                                  malloc.c:233)

                                  4 contexts (suppressed: 11 from 1)
                                  0 blocks.
                                  bytes allocated.
                                  with: -v
==26135== All heap blocks were freed -- no leaks are possible.
```

When all looks o.k. (8) – run valgrind

run:

```
$>  valgrind exp8

==26135== Invalid read of size 1
==26135==    at 0x401E208: strlen (mc_replace_strmem.c:246)
==26135==    by 0x405F0C7: vfprintf (vfprintf.c:1535)
==26135==    by 0x4064C72: printf (printf.c:34)
==26135==    by 0x8048410: main (exp8.c:10)
==26135==  Address 0x413C028 is 0 bytes inside a block of size 1,024
   free'd
==26135==    at 0x401CFA5: free (vg_replace_malloc.c:233)
==26135==    by 0x80483FD: main (exp8.c:9)
```