CIS212 Lab Week #5

Valgrind

UNIVERSITY OF
OREGON

Valgrind memory error detection

- Invalid read/write
- Uninitialized Values
- Memory leaks
- Illegal frees

And more but we'll just look at these, for more info see `man valgrind' or go to:

http://valgrind.org/docs/manual/mc-manual.html#mc-manual.badrw

Invalid read/write

This happens when valgind detects that your program is reading from or writing to part of the memory that you don’t have access to.

Often results in segmentation fault

Why?

Invalid read/write

Example:

```
int a = 5;
printf (“%s\n”, a);
```

This is an invalid read.

Why?

Uninitialized variables

This happens when your program declares a variable but doesn't initialize it to a value. And then uses that variable somewhere

Example:

```
int a;
printf ("%d\n", a)
```

Why?

Memory leak

This happens when your program allocates dynamic (heap) memory, but doesn't de-allocate that memory by the end of program:

Example:

```
int main () {
    char *c = malloc (10);
}
```

Why?

Invalid free

This happens when your program tries to free (de-allocate) memory that is not your programs to free, or you have already freed it, or it is not heap memory.

Example:

```
char *a = “hello world”;
free (a);
```

Why?