64-bit Differences

Sample Code

sizes.c

```c
#include <stdio.h>  /* printf   */
#include <stdint.h> /* intptr_t */

int main(void)
{
  int i = 123;       /* sizeof(int) depends on architecture */
  int *p1 = &i;      /* works on all architectures          */

  intptr_t j = 456;  /* intptr_t is a pointer-sized int */
  intptr_t *p2 = &j; /* works on all architectures      */

  i = (int)p1;       /* Probably works on 32-bit, may not on 64-bit */
  j = (intptr_t)p2;  /* Will work on either architecture             */

  printf("sizeof(short)     : %lu\n", sizeof(short));
  printf("sizeof(int)       : %lu\n", sizeof(int));
  printf("sizeof(long)      : %lu\n", sizeof(long));
  printf("sizeof(long long) : %lu\n", sizeof(long long));
  printf("sizeof(void*)     : %lu\n", sizeof(void*));
  printf("sizeof(size_t)    : %lu\n", sizeof(size_t));
  printf("sizeof(intptr_t)  : %lu\n", sizeof(intptr_t));

  return 0;
}
```

Outputs:

32-bit Windows/Linux/Cygwin

```
sizeof(short)     : 2
sizeof(int)       : 4
sizeof(long)      : 4
sizeof(long long) : 8
sizeof(void*)     : 4
sizeof(size_t)    : 4
sizeof(intptr_t)  : 4
```

64-bit Windows (LLP64)

```
sizeof(short)     : 2
sizeof(int)       : 4
sizeof(long)      : 4
sizeof(long long) : 8
sizeof(void*)     : 8
sizeof(size_t)    : 8
sizeof(intptr_t)  : 8
```

64-bit Linux/Mac OS X (LP64)

```
sizeof(short)     : 2
sizeof(int)       : 4
sizeof(long)      : 8
sizeof(long long) : 8
sizeof(void*)     : 8
sizeof(size_t)    : 8
sizeof(intptr_t)  : 8
```

Warning:

```
sizes.c:12:7: warning: cast from pointer
to integer of different size [-Wpointer-
to-int-cast]
```