**Format**  |  **Computation**  
---|---
`addl` `Src, Dest`  |  `Dest = Dest + Src`
`sbl` `Src, Dest`  |  `Dest = Dest - Src`
`imull` `Src, Dest`  |  `Dest = Dest * Src`
`sall` `Src, Dest`  |  `Dest = Dest << Src`
`sarl` `Src, Dest`  |  `Dest = Dest >> Src`
`shrl` `Src, Dest`  |  `Dest = Dest >> Src`
`xorl` `Src, Dest`  |  `Dest = Dest ^ Src`
`andl` `Src, Dest`  |  `Dest = Dest & Src`
`orl` `Src, Dest`  |  `Dest = Dest | Src`

**Bias**  
\[ Bias = 2^{k-1} - 1 \]

Big Endian: Sun, PPC Mac, Internet
- Least significant byte has highest address

Little Endian: x86
- Least significant byte has lowest address

---

**Specific Cases of Alignment (IA32)**

- **1 byte:** `char`, ...
  - no restrictions on address

- **2 bytes:** `short`, ...
  - lowest 1 bit of address must be 0₂

- **4 bytes:** `int, float, char *`, ...
  - lowest 2 bits of address must be 00₂

- **8 bytes:** `double, ...`
  - Windows (and most other OS’s & instruction sets):
    - lowest 3 bits of address must be 000₂
  - Linux:
    - lowest 2 bits of address must be 00₂
    - i.e., treated the same as a 4-byte primitive data type

- **12 bytes:** `long double`
  - Windows, Linux:
    - lowest 2 bits of address must be 00₂
    - i.e., treated the same as a 4-byte primitive data type

**Specific Cases of Alignment (x86-64)**

- **1 byte:** `char`, ...
  - no restrictions on address

- **2 bytes:** `short, ...`
  - lowest 1 bit of address must be 0₂

- **4 bytes:** `int, float, ...`
  - lowest 2 bits of address must be 00₂

- **8 bytes:** `double, char *`, ...
  - Windows & Linux:
    - lowest 3 bits of address must be 000₂

- **16 bytes:** `long double`
  - Linux:
    - lowest 3 bits of address must be 000₂
    - i.e., treated the same as a 8-byte primitive data type
<table>
<thead>
<tr>
<th>Register</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%rax</td>
<td>Return value</td>
</tr>
<tr>
<td>%rbx</td>
<td>Callee saved</td>
</tr>
<tr>
<td>%rcx</td>
<td>4th argument</td>
</tr>
<tr>
<td>%rdx</td>
<td>3rd argument</td>
</tr>
<tr>
<td>%rsi</td>
<td>2nd argument</td>
</tr>
<tr>
<td>%rdi</td>
<td>1st argument</td>
</tr>
<tr>
<td>%rbp</td>
<td>Callee saved</td>
</tr>
<tr>
<td>%rsp</td>
<td>Stack pointer</td>
</tr>
<tr>
<td>%r8</td>
<td>5th argument</td>
</tr>
<tr>
<td>%r9</td>
<td>6th argument</td>
</tr>
<tr>
<td>%r10</td>
<td>Callee saved</td>
</tr>
<tr>
<td>%r11</td>
<td>Used for linking</td>
</tr>
<tr>
<td>%r12</td>
<td>Unused for C</td>
</tr>
<tr>
<td>%r13</td>
<td>Callee saved</td>
</tr>
<tr>
<td>%r14</td>
<td>Callee saved</td>
</tr>
<tr>
<td>%r15</td>
<td>Callee saved</td>
</tr>
</tbody>
</table>