

# Link 7.B Static Linking

Young W. Lim

2019-01-28 Mon

- 1 Based on
- 2 Static Linking Examples
- 3 relocation information

"Self-service Linux: Mastering the Art of Problem Determination",

Mark Wilding

"Computer Architecture: A Programmer's Perspective",

Bryant & O'Hallaron

I, the copyright holder of this work, hereby publish it under the following licenses: GNU head Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled GNU Free Documentation License.

CC BY SA This file is licensed under the Creative Commons Attribution ShareAlike 3.0 Unported License. In short: you are free to share and make derivative works of the file under the conditions that you appropriately attribute it, and that you distribute it only under a license compatible with this one.

# Compiling 32-bit program on 64-bit gcc

- `gcc -v`
- `gcc -m32 t.c`
- `sudo apt-get install gcc-multilib`
- `sudo apt-get install g++-multilib`
- `gcc-multilib`
- `g++-multilib`
- `gcc -m32`
- `objdump -m i386`

# TOC: Static Linking Examples

- 1 `addvec.c` and `mutvec.c`
- 2 `libvector.a`
- 3 `main.c`
- 4 `p`

## addvec.c and multvec.c

```
/*::::: addvec.c ::::::::::::::::::::*/  
void addvec(int *x, int *y, int *z, int n)  
{  
    int i;  
  
    for (i=0; i<n; i++)  
        z[i] = x[i] + y[i];  
  
}
```

```
/*::::: multvec.c ::::::::::::::::::::*/  
void multvec(int *x, int *y, int *z, int n)  
{  
    int i;  
  
    for (i=0; i<n; i++)  
        z[i] = x[i] * y[i];  
  
}
```

```
/*::::: vector.h ::::::::::::::::::::*/  
void addvec(int *x, int *y, int *z, int n);  
void multvec(int *x, int *y, int *z, int n);
```

```
/*::::: main.c ::::::::::::::::::::*/  
#include <stdio.h>  
#include "vector.h"  
  
int x[2] = { 1, 2};  
int y[2] = { 3, 4};  
int z[2];  
  
int main() {  
  
    addvec(x, y, z, 2);  
    printf("z= [%d %d]\n", z[0], z[1]);  
  
}
```

# compiling commands

- ```
gcc -g -m32 -Wall -c addvec.c
gcc -g -m32 -Wall -c multvec.c
ar rcs libvector.a addvec.o multvec.o

gcc -g -m32 -Wall -c main.c
gcc -g -m32 -static main.o -L. -lvector -o staticcp
```



# analyzing commands

- `objdump -d addvec.o ; objdump -d staticp`  
`objdump -dr addvec.o ; objdump -dr staticp`  
`objdump -D addvec.o ; objdump -D staticp`
- `readelf -r addvec.o`  
`readelf -r libvector.a`  
`readelf -r staticp`
- `readelf -S staticp (--sections)`  
`readelf -s staticp (--symbols)`

# relocation information in addvec.o

```
objdump -dr addvec.o
```

```
7:  e8 fc ff ff ff          call    8 <addvec+0x8>
      8: R_386_PC32    __x86.get_pc_thunk.ax
c:  05 01 00 00 00          add     $0x1,%eax
      d: R_386_GOTPC  _GLOBAL_OFFSET_TABLE_
```

```
readelf -r addvec.o
```

| Offset   | Info     | Type        | Sym.Value | Sym. Name             |
|----------|----------|-------------|-----------|-----------------------|
| 00000008 | 00001002 | R_386_PC32  | 00000000  | __x86.get_pc_thunk.ax |
| 0000000d | 0000110a | R_386_GOTPC | 00000000  | _GLOBAL_OFFSET_TABLE_ |

# relocation information in multvec.o

```
objdump -dr multvec.o
```

```
7:  e8 fc ff ff ff          call    8 <multvec+0x8>
                               8: R_386_PC32    __x86.get_pc_thunk.ax
c:  05 01 00 00 00          add    $0x1,%eax
                               d: R_386_GOTPC  _GLOBAL_OFFSET_TABLE_
```

```
readelf -r multvec.o
```

| Offset   | Info     | Type        | Sym.Value | Sym. Name             |
|----------|----------|-------------|-----------|-----------------------|
| 00000008 | 00001002 | R_386_PC32  | 00000000  | __x86.get_pc_thunk.ax |
| 0000000d | 0000110a | R_386_GOTPC | 00000000  | _GLOBAL_OFFSET_TABLE_ |

# relocation information in main.o (1)

```
objdump -dr main.o
```

```
f:   e8 fc ff ff ff          call   10 <main+0x10>
                                10: R_386_PC32   __x86.get_pc_thunk.bx
14:   81 c3 02 00 00 00      add    $0x2,%ebx
                                16: R_386_GOTPC   _GLOBAL_OFFSET_TABLE_
1c:   8b 83 00 00 00 00      mov    0x0(%ebx),%eax
                                1e: R_386_GOT32X   z
23:   8d 83 00 00 00 00      lea   0x0(%ebx),%eax
                                25: R_386_GOTOFF   y
2a:   8d 83 00 00 00 00      lea   0x0(%ebx),%eax
                                2c: R_386_GOTOFF   x
```

```
readelf -r main.o
```

| Offset   | Info     | Type         | Sym.Value | Sym. Name             |
|----------|----------|--------------|-----------|-----------------------|
| 00000010 | 00001402 | R_386_PC32   | 00000000  | __x86.get_pc_thunk.bx |
| 00000016 | 0000150a | R_386_GOTPC  | 00000000  | _GLOBAL_OFFSET_TABLE_ |
| 0000001e | 0000122b | R_386_GOT32X | 00000004  | z                     |
| 00000025 | 00001109 | R_386_GOTOFF | 00000008  | y                     |
| 0000002c | 00001009 | R_386_GOTOFF | 00000000  | x                     |

# relocation information in main.o (2)

```
objdump -dr main.o
```

```
31:  e8 fc ff ff ff          call   32 <main+0x32>
                               32: R_386_PLT32 addvec
39:  8b 83 00 00 00 00      mov    0x0(%ebx),%eax
                               3b: R_386_GOT32X      z
42:  8b 83 00 00 00 00      mov    0x0(%ebx),%eax
                               44: R_386_GOT32X      z
4f:  8d 83 00 00 00 00      lea   0x0(%ebx),%eax
                               51: R_386_GOTOFF      .rodata
56:  e8 fc ff ff ff          call   57 <main+0x57>
                               57: R_386_PLT32 printf
```

```
readelf -r main.o
```

| Offset   | Info     | Type         | Sym.Value | Sym. Name             |
|----------|----------|--------------|-----------|-----------------------|
| 00000010 | 00001402 | R_386_PC32   | 00000000  | __x86.get_pc_thunk.bx |
| 00000016 | 0000150a | R_386_GOTPC  | 00000000  | _GLOBAL_OFFSET_TABLE_ |
| 0000001e | 0000122b | R_386_GOT32X | 00000004  | z                     |
| 00000025 | 00001109 | R_386_GOTOFF | 00000008  | y                     |
| 0000002c | 00001009 | R_386_GOTOFF | 00000000  | x                     |
| 00000032 | 00001604 | R_386_PLT32  | 00000000  | addvec                |
| 0000003b | 0000122b | R_386_GOT32X | 00000004  | z                     |
| 00000044 | 0000122b | R_386_GOT32X | 00000004  | z                     |