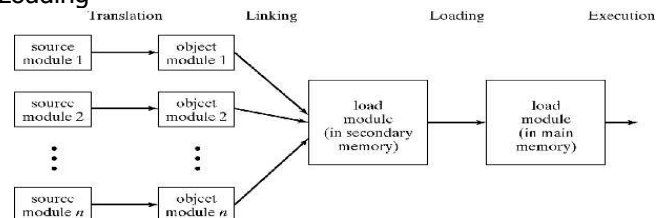


Miscellaneous Memory Management topics

Preparing Program for Execution

- Discussion taken from "Operating Systems Principles", Bic and Shaw, Prentice Hall, 2003.
- Program Transformations
 - Translation (Compilation)
 - Linking
 - Loading

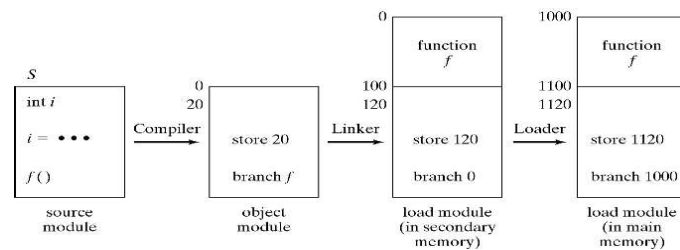


Address Binding

- Assign Physical Addresses = Relocation
- Static binding
 - Programming time
 - Compilation time
 - Linking time
 - Loading time
- Dynamic binding
 - Execution time

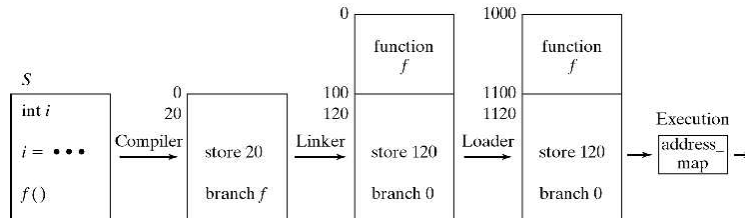
Static Address Binding

Static Binding = At Programming, Compilation, Linking, and/or Loading Time



Dynamic Address Binding

Dynamic Binding = At Execution Time



Address Binding

- How to implement dynamic binding
 - Perform for each address at run time:
 $pa = \text{address_map}(la)$
 - Simplest form of `address_map`:
Relocation Register: $pa = la + RR$
 - More general form:
Page/Segment Table

Third-chance algorithm

- Second chance algorithm does not distinguish between read and write access
- Write access more expensive
- Give modified pages a third chance:
 - *u*-bit set at every reference (read and write)
 - *w*-bit set at write reference
 - to select a page, cycle through frames, resetting bits, until ***uw*==00**:

<u>uw</u> → <u>uw</u>	
1 1	0 1
1 0	0 0
0 1	0 0* (remember modification)
0 0	select

Third-chance algorithm

- Read→10→00→Select
Write→11→01→00*→Select

...	0	1	2	3	4	5	6	7	8	9	10	.
...	c	a*	d	b*	e	b	a*	b	c	d	.	
>a/10	>a/10	>a/11	>a/11	>a/11	a/00*	a/00*	a/11	a/11	>a/11	a/00*	.	
... b/10	b/10	b/10	b/10	b/11	b/00*	b/10*	b/10*	b/10*	b/10*	b/10*	d/10	.
... c/10	c/10	c/10	c/10	c/10	e/10	e/10	e/10	e/10	e/10	>e/00	.	
... d/10	>d/10	d/10	d/10	d/10	>d/00	>d/00	>d/00	>d/00	>d/00	c/10	c/00	.
... IN					e					c	d	
... OUT					c					d	b	