

Homework 4 – Heap Assignment

Name: Goutami Padmanabhan

UTA ID: 1001669338

Benchmarks: A discussion on how you benchmarked your library and what the results were.

Which heap management strategy does the best job of reusing free blocks? **Best Fit**

Which one is the worst? **Next Fit**

Which heap management strategy requires the least amount of heap space? **Best Fit**

Which one is the worst? **Worst Fit**

Which heap management strategy allows for the most splits? **Worst Fit**

most coalescing? **Worst Fit**

least splits? **Best Fit**

least coalescing? **Best Fit**

Which heap management strategy was the fastest? **Next Fit**

slowest? **Best Fit**

Consider the benchmark programs.

Which one requires the most mallocs? **Next Fit**

frees? **Next Fit**

Which one requests the most amount of space? **Worst Fit**

Which one requires the largest heap? **Worst Fit**

Which heap management strategy suffers the most from fragmentation (what type)? **First Fit – Internal fragmentation**

Which heap management strategy is the best? **Next Fit**

Summary:

First Fit:

Concept: First fit allocates a process to a block which is the first enough block that is able to accommodate the process. It starts searching from the first block.

Advantages: Simple and fast search

Disadvantages: Internal fragmentation at the beginning of the search. Causes External Fragmentation also. Wastage of space.

Next Fit:

Concept: Next Fit allocates a process to a block which is the next enough block that is able to accommodate the process. It starts searching from the last block it left previously.

Advantages: Avoids internal fragmentation by searching from where it ends last time. Fastest search.

Best Fit:

Concept: Best fit keeps searching for the minimum block size available that could accommodate a process and allocates it.

Advantages: Space is efficiently utilized. Memory utilization is the best.

Disadvantages: It consumes a lot of time to search for the best block to accommodate. Slow search. As a result of this, performance is less.

Worst Fit: Worst fit keeps searching for the maximum block size available that could accommodate a process and allocates it.

Advantages: It reduces the small holes in memory.

Disadvantages: Larger block sizes are occupied at the beginning. If a process requires large space at the end, it is incapable of providing the space.

B1	
B2	2K
B3	5K
B4	
B5	
B6	13K
B7	10K
B8	15K
B9	20K
B10	25K

First Fit -

<u>Process size</u>	<u>Block No.</u>
5K	B3
10K	B6
2K	Not allocated

Next Fit

- Assume last execution was till B6

<u>Process size</u>	<u>Block No.</u>
20K	B9
15K	Not allocated

Best Fit

<u>Process size</u>	<u>Block No.</u>
3K	B3
10K	B7

Worst Fit

<u>Process size</u>	<u>Block No.</u>
3K	B10
10K	B9

Results for Analysis:

First Fit

Test1.c

```
Running test 1 to test a simple malloc and free

heap management statistics
mallocs:      1
frees:        1
reuses:       0
grows:        1
splits:       0
coalesces:    0
blocks:       1
requested:    65536
max heap:     65536
```

Test2.c

```
Running test 2 to exercise malloc and free

heap management statistics
mallocs:      1026
frees:        514
reuses:       0
grows:        1025
splits:       0
coalesces:    1
blocks:       1024
requested:    1179648
max heap:     1114112
```

Test3.c

```
Running test 3 to test coalesce

heap management statistics
mallocs:      3
frees:        3
reuses:       0
grows:        3
splits:       0
coalesces:    0
blocks:       3
requested:    4448
max heap:     4448
```

Test4.c

```
Running test 4 to test a block split and reuse

heap management statistics
mallocs:      2
frees:        2
reuses:       0
grows:        1
splits:       1
coalesces:    1
blocks:       1
requested:    3072
max heap:     2048
```

Ffnf.c

```
First fit should pick this one: 0x1b4c2018
Next fit should pick this one: 0x1b4c3ba0
Chosen address: 0x1b4c2018

heap management statistics
mallocs:      5
frees:        3
reuses:       0
grows:        3
splits:       0
coalesces:    0
blocks:       3
requested:    15000
max heap:     8000
```

Bench1.c

```
heap management statistics
mallocs:      2049
frees:        513
reuses:       0
grows:        1025
splits:       1023
coalesces:    0
blocks:       2048
requested:    4171424
max heap:     4167328
```

Bench2.c

```
heap management statistics
mallocs:      4609
frees:        4609
reuses:       0
grows:        2880
splits:       1727
coalesces:    2560
blocks:       2047
requested:    15990784
max heap:     12845056
```

Next Fit:

Test1.c

```
Running test 1 to test a simple malloc and free

heap management statistics
mallocs:      1
frees:        1
reuses:       0
grows:        1
splits:       0
coalesces:    0
blocks:       1
requested:    65536
max heap:     65536
```

Test2.c

```
Running test 2 to exercise malloc and free

heap management statistics
mallocs:      1026
frees:        514
reuses:       0
grows:        1025
splits:       0
coalesces:    1
blocks:       1024
requested:    1179648
max heap:     1114112
```

Test3.c

```
Running test 3 to test coalesce

heap management statistics
mallocs:      3
frees:        3
reuses:       0
grows:        3
splits:       0
coalesces:    0
blocks:       3
requested:    4448
max heap:     4448
```


Test4.c

```
Running test 4 to test a block split and reuse

heap management statistics
mallocs:      2
frees:        2
reuses:       0
grows:        1
splits:       1
coalesces:    1
blocks:       1
requested:    3072
max heap:     2048
```

Ffnf.c

```
First fit should pick this one: 0xf860018
Next fit should pick this one: 0xf861ba0
Chosen address: 0xf860018

heap management statistics
mallocs:      5
frees:        3
reuses:       0
grows:        3
splits:       0
coalesces:    0
blocks:       3
requested:    15000
max heap:     8000
```

Bench1.c

```
heap management statistics
mallocs:      2049
frees:        513
reuses:       0
grows:        1024
splits:       1023
coalesces:    0
blocks:       2047
requested:    4114496
max heap:     4110396
```

Bench2.c

```
heap management statistics
mallocs:      4609
frees:        4609
reuses:       0
grows:        2880
splits:       1727
coalesces:    2560
blocks:       2047
requested:    15990784
max heap:     12845056
```

Best Fit:

Test1.c

```
Running test 1 to test a simple malloc and free
```

```
heap management statistics
```

```
mallocs:      1
frees:        1
reuses:       0
grows:        1
splits:       0
coalesces:    0
blocks:       1
requested:    65536
max heap:     65536
```

Test2.c

```
Running test 2 to exercise malloc and free
```

```
heap management statistics
```

```
mallocs:      1026
frees:        514
reuses:       0
grows:        1025
splits:       0
coalesces:    1
blocks:       1024
requested:    1179648
max heap:     1114112
```

Test3.c

```
Running test 3 to test coalesce

heap management statistics
mallocs:      3
frees:        3
reuses:       0
grows:        3
splits:       0
coalesces:    0
blocks:       3
requested:    4448
max heap:     4448
```

Test4.c

```
Running test 4 to test a block split and reuse

heap management statistics
mallocs:      2
frees:        2
reuses:       0
grows:        1
splits:       1
coalesces:    1
blocks:       1
requested:    3072
max heap:     2048
```

Bfwf.c

```
Worst fit should pick this one: 0x68d5018  
Best fit should pick this one: 0x68e5030  
Chosen address: 0x68e5030
```

```
heap management statistics  
mallocs:      3  
frees:        2  
reuses:       0  
grows:        2  
splits:       1  
coalesces:    0  
blocks:       3  
requested:    72536  
max heap:     71536
```

Bench1.c

```
heap management statistics  
mallocs:      2049  
frees:        513  
reuses:       0  
grows:        1024  
splits:       1023  
coalesces:    0  
blocks:       2047  
requested:    4192868  
max heap:     4188768
```

Bench2.c

```
heap management statistics
mallocs:      4609
frees:        4609
reuses:       0
grows:        2880
splits:       1727
coalesces:    2560
blocks:       2047
requested:    15990784
max heap:     12845056
```

Worst Fit:

Test1.c

```
Running test 1 to test a simple malloc and free

heap management statistics
mallocs:      1
frees:        1
reuses:       0
grows:        1
splits:       0
coalesces:    0
blocks:       1
requested:    65536
max heap:     65536
```

Test2.c

```
Running test 2 to exercise malloc and free

heap management statistics
mallocs:      1026
frees:        514
reuses:       0
grows:        1025
splits:       0
coalesces:    1
blocks:       1024
requested:    1179648
max heap:     1114112
```

Test3.c

```
Running test 3 to test coalesce

heap management statistics
mallocs:      3
frees:        3
reuses:       0
grows:        3
splits:       0
coalesces:    0
blocks:       3
requested:    4448
max heap:     4448
```

Test4.c

```
Running test 4 to test a block split and reuse

heap management statistics
mallocs:      2
frees:        2
reuses:       0
grows:        1
splits:       1
coalesces:    1
blocks:       1
requested:    3072
max heap:     2048
```

Bfwf.c

```
Worst fit should pick this one: 0xd56018
Best fit should pick this one: 0xd66030
Chosen address: 0xd56018

heap management statistics
mallocs:      3
frees:        2
reuses:       0
grows:        2
splits:       1
coalesces:    0
blocks:       3
requested:    72536
max heap:     71536
```


Bench1.c

```
heap management statistics
mallocs:      2049
frees:        513
reuses:       0
grows:        1024
splits:       1023
coalesces:    0
blocks:       2047
requested:    4093728
max heap:     4089628
```

Bench2.c

```
heap management statistics
mallocs:      4609
frees:        4609
reuses:       0
grows:        2880
splits:       1727
coalesces:    2560
blocks:       2047
requested:    15990784
max heap:     12845056
```