King Saud University

كلية علوم الحاسب والمعلومات
قسم تقنية المعلومات

College of Computer and Information Sciences
Department of Information Technology

Programming Assignment2
Memory Fragmentation Simulation
CSC 227 Semester-2, 1446H

Student names and IDs:

Student's Names	Student's IDs	Section
Maha Albakr	444201108	78287
Hatoun Ibrahim Almogherah	444203015	78287
Rama Khalid Alomair	444200662	78287
Walaa Saif Aleslam	444200088	78287

Supervised By: Dr. Rabia Jafri

Task Distribution:

Student's Names	Task
Hatoun Ibrahim Almogherah	- Design the MemoryBlock class - Implement initializeMemory() - Create memory blocks with proper start/end addresses (MemoryBlock class, initializeMemory()) - Implement main() loop and menu
Maha Albakr	- Handle both initial and detailed report (printInitialReport(), printReport()) - Implement main() loop and menu
Rama Khalid Alomair	- Handle memory allocation strategies - Implement First-Fit, Best-Fit, and Worst-Fit - Calculate internal fragmentation allocateMemory()
Walaa Saif Aleslam	- Handle deallocation - Search by process ID - Handle invalid PID case deallocateMemory()

Screen shots showing sample input/output:

Test Case 1: First-Fit Allocation Success

```
Enter the total number of blocks: 3
Enter the size of each block in KB: 100 200 300
Enter allocation strategy (1=First-Fit, 2=Best-Fit, 3=Worst-Fit): 1

Memory blocks:
=========================================
Block#    Size      Start-End      Status
=========================================
0         100       0-99           free
1         200       100-299        free
2         300       300-599        free
=========================================

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
=========================================
Enter your choice: 1
Enter the process ID and size of process: P1 150
P1 allocated at address 100, internal fragmentation: 50
```

Test Case 2: Best-Fit Allocation Success

```
Enter the total number of blocks: 4
Enter the size of each block in KB: 300 200 100 400
Enter allocation strategy (1=First-Fit, 2=Best-Fit, 3=Worst-Fit): 2

Memory blocks:
=========================================
Block#    Size      Start-End      Status
=========================================
0         300       0-299          free
1         200       300-499        free
2         100       500-599        free
3         400       600-999        free
=========================================

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
=========================================
Enter your choice: 1
Enter the process ID and size of process: P1 60
P1 allocated at address 500, internal fragmentation: 40
```

3

Test Case 3: Worst-Fit Allocation Success

```
Enter the total number of blocks: 4
Enter the size of each block in KB: 300 200 100 400
Enter allocation strategy (1=First-Fit, 2=Best-Fit, 3=Worst-Fit): 3

Memory blocks:
========================================
Block#   Size      Start-End      Status
========================================
0        300       0-299          free
1        200       300-499        free
2        100       500-599        free
3        400       600-999        free
========================================

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
========================================
Enter your choice: 1
Enter the process ID and size of process: P1 60
P1 allocated at address 600, internal fragmentation: 340
```

Test Case 4: Allocation Failure

```
Enter the total number of blocks: 2
Enter the size of each block in KB: 100 200
Enter allocation strategy (1=First-Fit, 2=Best-Fit, 3=Worst-Fit): 2

Memory blocks:
========================================
Block#   Size      Start-End      Status
========================================
0        100       0-99           free
1        200       100-299        free
========================================

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
========================================
Enter your choice: 1
Enter the process ID and size of process: P1 300
Allocation failed: no suitable block found.
```

Test Case 5: Deallocate and Reallocate

```
Enter the total number of blocks: 3
Enter the size of each block in KB: 100 200 300
Enter allocation strategy (1=First-Fit, 2=Best-Fit, 3=Worst-Fit): 1

Memory blocks:
==============================================
Block#    Size      Start-End      Status
==============================================
0         100       0-99           free
1         200       100-299        free
2         300       300-599        free
==============================================

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
==============================================
Enter your choice: 1
Enter the process ID and size of process: P1 150
P1 allocated at address 100, internal fragmentation: 50

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
==============================================
Enter your choice: 2
Enter the process ID to deallocate: P1
Process P1 deallocated.

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
==============================================
Enter your choice: 1
Enter the process ID and size of process: P2 100
P2 allocated at address 0, internal fragmentation: 0
```

Test Case 6: Internal Fragment Accuracy

```
Enter the total number of blocks: 3
Enter the size of each block in KB: 300 200 100
Enter allocation strategy (1=First-Fit, 2=Best-Fit, 3=Worst-Fit): 2

Memory blocks:
==========================================
Block#   Size      Start-End      Status
==========================================
0        300       0-299          free
1        200       300-499        free
2        100       500-599        free
==========================================

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
==========================================
Enter your choice: 1
Enter the process ID and size of process: P1 90
P1 allocated at address 500, internal fragmentation: 10

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
==========================================
Enter your choice: 1
Enter the process ID and size of process: P2 190
P2 allocated at address 300, internal fragmentation: 10

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
==========================================
Enter your choice: 3

Memory blocks:
============================================================================
Block#   Size      Start-End      Status        ProcessID      Fragment
============================================================================
0        300       0-299          free          Null           0
1        200       300-499        allocated      P2             10
2        100       500-599        allocated      P1             10
============================================================================
```

Test Case 7: Invalid Deallocation

```
Enter the total number of blocks: 2
Enter the size of each block in KB: 100 200
Enter allocation strategy (1=First-Fit, 2=Best-Fit, 3=Worst-Fit): 1

Memory blocks:
==========================================
Block#    Size      Start-End      Status
==========================================
0         100       0-99           free
1         200       100-299        free
==========================================

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
==========================================
Enter your choice: 2
Enter the process ID to deallocate: X99
Process not found.
```

Test Case 8: Full Report After Allocations

```
Enter the total number of blocks: 3
Enter the size of each block in KB: 150 250 100
Enter allocation strategy (1=First-Fit, 2=Best-Fit, 3=Worst-Fit): 1

Memory blocks:
========================================
Block#    Size     Start-End      Status
========================================
0         150      0-149          free
1         250      150-399        free
2         100      400-499        free
========================================

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
========================================
Enter your choice: 1
Enter the process ID and size of process: P1 100
P1 allocated at address 0, internal fragmentation: 50

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
========================================
Enter your choice: 1
Enter the process ID and size of process: P2 240
P2 allocated at address 150, internal fragmentation: 10

1) Allocates memory blocks
2) De-allocates memory blocks
3) Print report about the current state of memory and internal Fragmentation
4) Exit
========================================
Enter your choice: 3

Memory blocks:
========================================================================
Block#    Size     Start-End      Status        ProcessID      Fragment
========================================================================
0         150      0-149          allocated     P1             50
1         250      150-399        allocated     P2             10
2         100      400-499        free          Null           0
========================================================================
```