



Ain Shams University
Faculty of Engineering

Software Manual

Assignment 2: Memory Allocation

Names	Mahmoud Hamdy Abd El-Gawwad Mohamed Mohamed El-Morsy
Section	3

Cairo , 2018

Step 1:

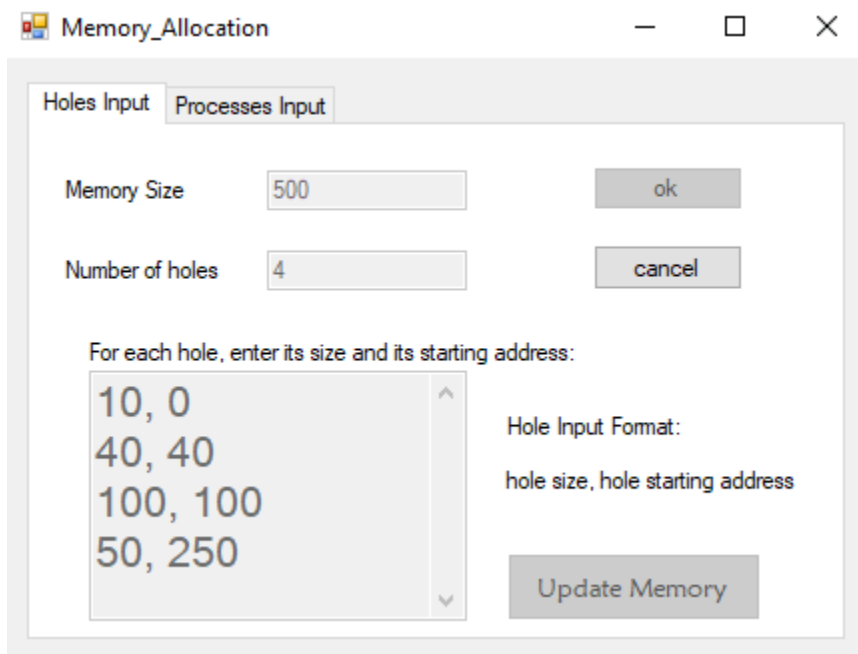
- In “Holes Input” tab, choose Memory Size and Number of Holes, then press the “ok” button.
- Note that you can not change those parameters except in the beginning, so, you can press the ”cancel” button in this tab as much as you want even if you pressed ”update Memory”.
- Enter holes data in the text box below then, press the “Update Memory” button.
- Now, your memory is ready.

The screenshot shows a window titled "Memory_Allocation" with two tabs: "Holes Input" and "Processes Input". The "Holes Input" tab is active. It contains two input fields: "Memory Size" with the value "500" and "Number of holes" with the value "4". To the right of these fields are "ok" and "cancel" buttons. A red circle with the number "1" is drawn around the "Memory Size" input field. Below these fields is a text box with the instruction "For each hole, enter its size and its starting address:". The text box contains four lines of input: "10, 0", "40, 40", "100, 100", and "50, 250". A red circle with the number "2" is drawn around the second line of input. To the right of the text box is the text "Hole Input Format:" followed by "hole size, hole starting address". Below the text box is an "Update Memory" button. A red checkmark is drawn next to the "Update Memory" button.

Hole Input Format:	
hole size	hole starting address
10	0
40	40
100	100
50	250

Step 2:

- In “Processes Input” tab, choose the format to write the processes data depending on whether you are allocating or de-allocating processes.
- In the “Allocation Algorithm” Group Box, check your requested algorithm and press the “Allocate/De-Allocate” button.
- Then, you can view the memory by pressing on “Show Memory” button.



The screenshot shows a window titled "Memory_Allocation" with standard Windows window controls (minimize, maximize, close). Inside the window, there are two tabs: "Holes Input" and "Processes Input". The "Processes Input" tab is currently selected. It contains the following elements:

- A "Memory Size" label followed by a text box containing the value "500" and an "ok" button.
- A "Number of holes" label followed by a text box containing the value "4" and a "cancel" button.
- A text instruction: "For each hole, enter its size and its starting address:".
- A multi-line text area containing the following text:

```
10, 0
40, 40
100, 100
50, 250
```
- A label "Hole Input Format:" followed by the text "hole size, hole starting address".
- An "Update Memory" button.

Memory_Allocation

Holes Input Processes Input

Process Input Format:
For Allocation: name,size
For De-Allocation: name

P1, 8
P2, 30
P3, 80
P4, 40

Allocation Algorithm

☒ First Fit
☐ Best Fit
☐ Worst Fit
☐ De-Allocation

Allocate / De-Allocate
Show Memory

Memory_Allocation

Holes Input Processes Input

Process Input Format:
For Allocation: name,size
For De-Allocation: name

P1, 8

Allocation Algorithm

☒ First Fit

Form2.cs [Design] Form1.cs [Design]

Assignment2_MemAllocation.Memory_Allc - btn_Allocation_Click(object sende

s.Generic;
odel;

Memory

P1 base: 0 size: 8 type: p	hole base: 8 size: 2 type: h	reserved base: 10 size: 30 type: r	P2 base: 40 size: 30 type: p	hole base: 70 size: 10 type: h	reserved base: 80 size: 20 type: r	P3 base: 100 size: 80 type: p	hole base: 180 size: 20 type: h
-------------------------------------	---------------------------------------	---	---------------------------------------	---	---	--	--

Close

- We can also return again by clicking the “close” button to re-(allocate/De-allocate) a process.
- To De-Allocate a process, write its name, check De-Allocation option, then press “Allocate/De-Allocate”.

Memory_Allocation

Holes Input Processes Input

Process Input Format:
For Allocation: name,size
For De-Allocation: name

P1

Allocation Algorithm

☐ First Fit
☐ Best Fit
☐ Worst Fit
☒ De-Allocation

Allocate / De-Allocate Show Memory

Memory

hole base: 0 size: 10 type: h	reserved base: 10 size: 30 type: r	P2 base: 40 size: 30 type: p	hole base: 70 size: 10 type: h	reserved base: 80 size: 20 type: r	P3 base: 100 size: 80 type: p	hole base: 180 size: 20 type: h	reserved base: 200 size: 50 type: r
--	---	---------------------------------------	---	---	--	--	--

Close

First Fit:

Memory_Allocation

Holes Input Processes Input

Process Input Format:
For Allocation: name,size
For De-Allocation: name

P1, 8
P2, 30
P3, 80
P4, 40

Allocation Algorithm

☒ First Fit
☐ Best Fit
☐ Worst Fit
☐ De-Allocation

Allocate / De-Allocate Show Memory

Memory

P1 base: 0 size: 8 type: p	hole base: 8 size: 2 type: h	reserved base: 10 size: 30 type: r	P2 base: 40 size: 30 type: p	hole base: 70 size: 10 type: h	reserved base: 80 size: 20 type: r	P3 base: 100 size: 80 type: p	hole base: 180 size: 20 type: h
-------------------------------------	---------------------------------------	---	---------------------------------------	---	---	--	--

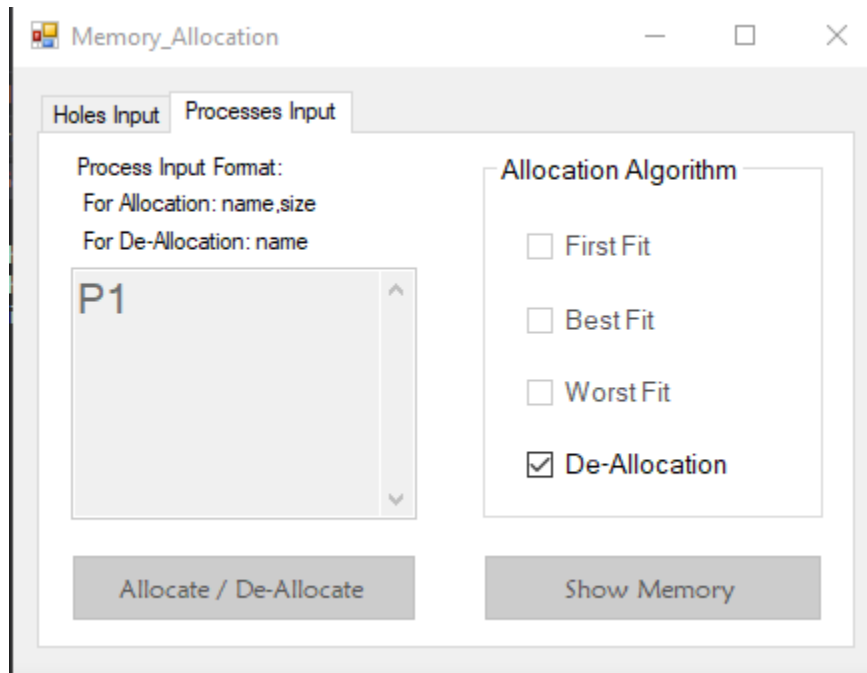
Close

Memory

hole base: 70 size: 10 type: h	reserved base: 80 size: 20 type: r	P3 base: 100 size: 80 type: p	hole base: 180 size: 20 type: h	reserved base: 200 size: 50 type: r	P4 base: 250 size: 40 type: p	hole base: 290 size: 10 type: h	reserved base: 300 size: 200 type: r
---	---	--	--	--	--	--	---

Close

De-Allocation:



Memory_Allocation

Holes Input Processes Input

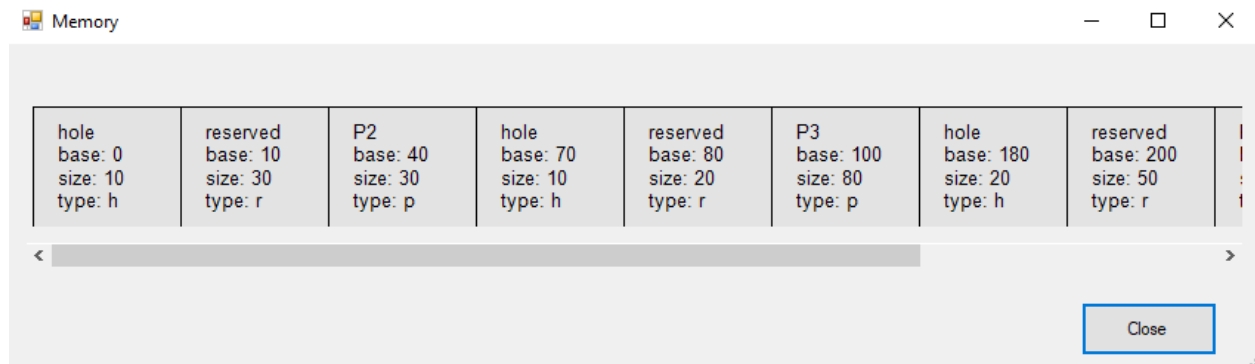
Process Input Format:
For Allocation: name,size
For De-Allocation: name

P1

Allocation Algorithm

- ☐ First Fit
- ☐ Best Fit
- ☐ Worst Fit
- ☒ De-Allocation

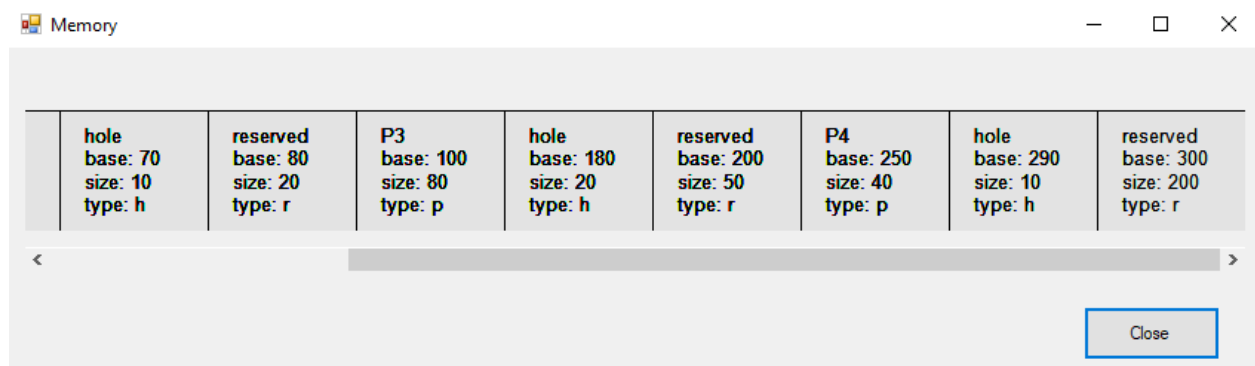
Allocate / De-Allocate Show Memory



Memory

hole base: 0 size: 10 type: h	reserved base: 10 size: 30 type: r	P2 base: 40 size: 30 type: p	hole base: 70 size: 10 type: h	reserved base: 80 size: 20 type: r	P3 base: 100 size: 80 type: p	hole base: 180 size: 20 type: h	reserved base: 200 size: 50 type: r
--	---	---------------------------------------	---	---	--	--	--

Close



Memory

hole base: 70 size: 10 type: h	reserved base: 80 size: 20 type: r	P3 base: 100 size: 80 type: p	hole base: 180 size: 20 type: h	reserved base: 200 size: 50 type: r	P4 base: 250 size: 40 type: p	hole base: 290 size: 10 type: h	reserved base: 300 size: 200 type: r
---	---	--	--	--	--	--	---

Close

Best Fit:

Memory_Allocation

Holes Input Processes Input

Process Input Format:
For Allocation: name,size
For De-Allocation: name

P1, 8
P2, 30
P3, 80
P4, 40

Allocation Algorithm

☐ First Fit
☒ Best Fit
☐ Worst Fit
☐ De-Allocation

Allocate / De-Allocate Show Memory

Memory

P1 base: 0 size: 8 type: p	hole base: 8 size: 2 type: h	reserved base: 10 size: 30 type: r	P2 base: 40 size: 30 type: p	hole base: 70 size: 10 type: h	reserved base: 80 size: 20 type: r	P3 base: 100 size: 80 type: p	hole base: 180 size: 20 type: h
-------------------------------------	---------------------------------------	---	---------------------------------------	---	---	--	--

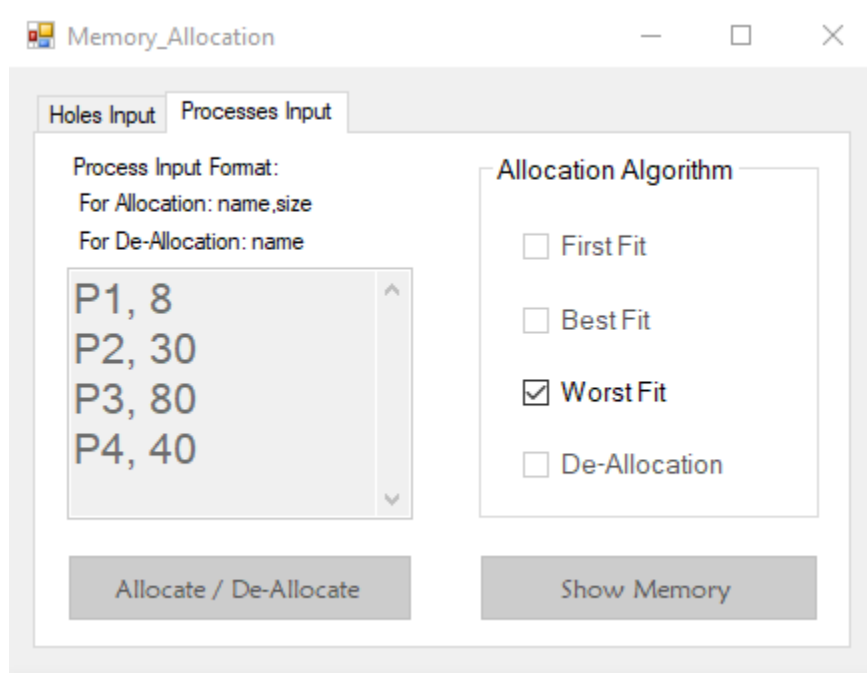
Close

Memory

hole base: 70 size: 10 type: h	reserved base: 80 size: 20 type: r	P3 base: 100 size: 80 type: p	hole base: 180 size: 20 type: h	reserved base: 200 size: 50 type: r	P4 base: 250 size: 40 type: p	hole base: 290 size: 10 type: h	reserved base: 300 size: 200 type: r
---	---	--	--	--	--	--	---

Close

Worst-Fit:



Memory_Allocation

Holes Input Processes Input

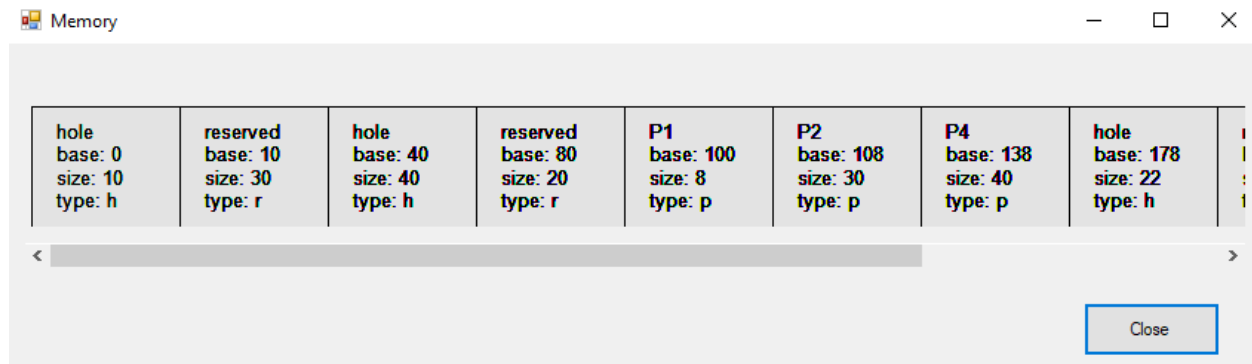
Process Input Format:
For Allocation: name,size
For De-Allocation: name

P1, 8
P2, 30
P3, 80
P4, 40

Allocation Algorithm

☐ First Fit
☐ Best Fit
☒ Worst Fit
☐ De-Allocation

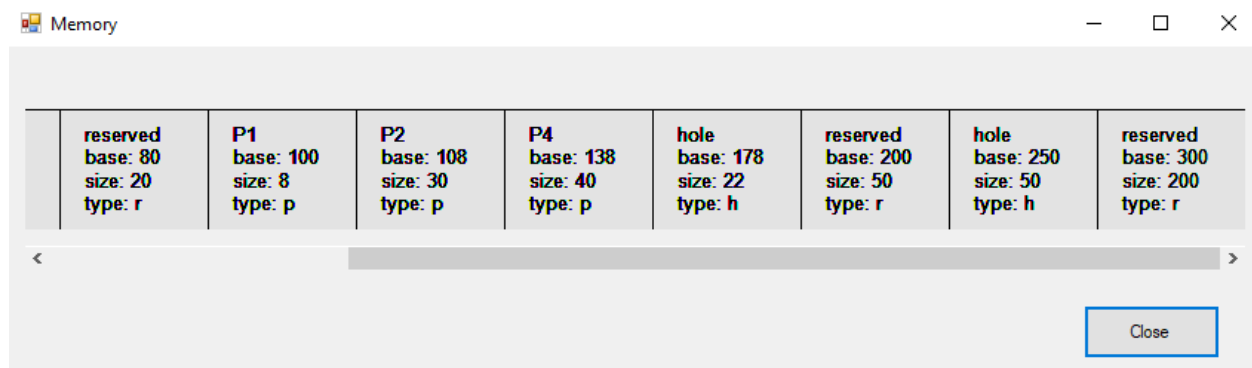
Allocate / De-Allocate Show Memory



Memory

hole base: 0 size: 10 type: h	reserved base: 10 size: 30 type: r	hole base: 40 size: 40 type: h	reserved base: 80 size: 20 type: r	P1 base: 100 size: 8 type: p	P2 base: 108 size: 30 type: p	P4 base: 138 size: 40 type: p	hole base: 178 size: 22 type: h
--	---	---	---	---------------------------------------	--	--	--

Close



Memory

reserved base: 80 size: 20 type: r	P1 base: 100 size: 8 type: p	P2 base: 108 size: 30 type: p	P4 base: 138 size: 40 type: p	hole base: 178 size: 22 type: h	reserved base: 200 size: 50 type: r	hole base: 250 size: 50 type: h	reserved base: 300 size: 200 type: r
---	---------------------------------------	--	--	--	--	--	---

Close