

## **Illustration 1**

Excel and Powerpoint

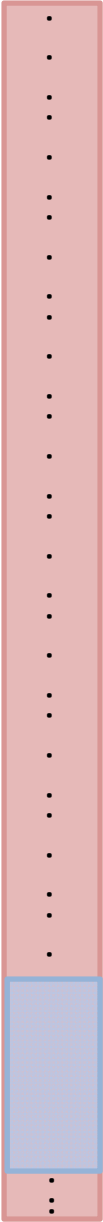
## **Illustration 2**

Java JVM

**Computer  
Memory**

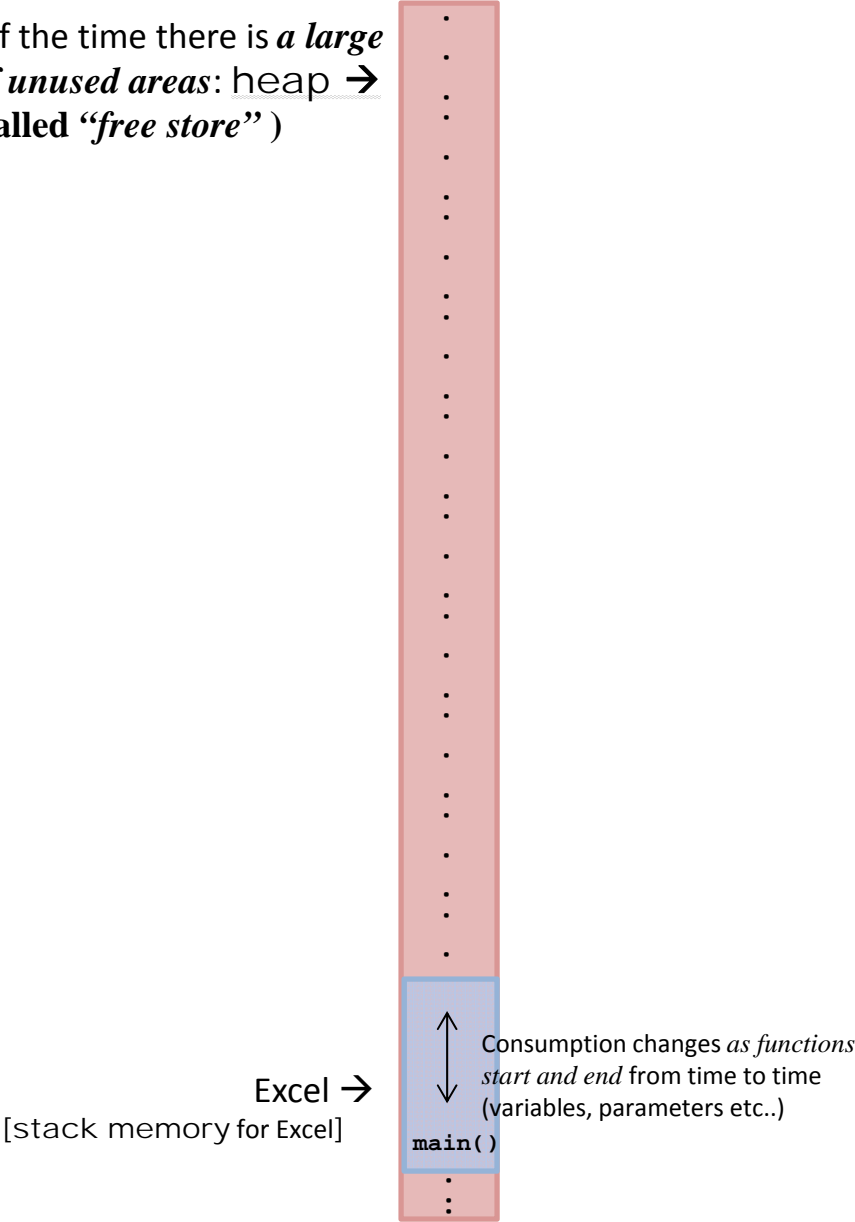
Most of the time there is *a large pool of unused areas*: heap → (also called “*free store*” )

Excel starts →  
[stack memory for Excel]



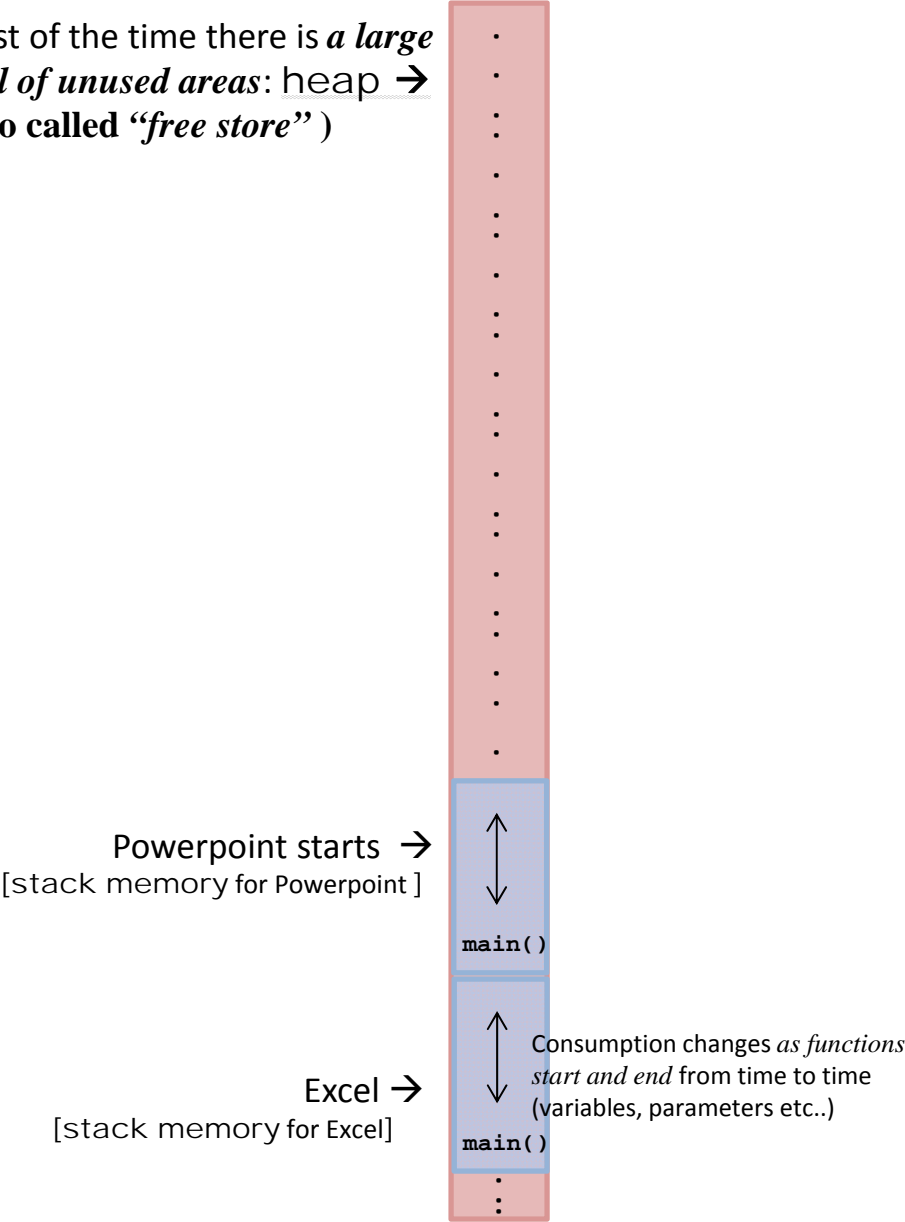
# Computer Memory

Most of the time there is *a large pool of unused areas*: heap → (also called “*free store*” )



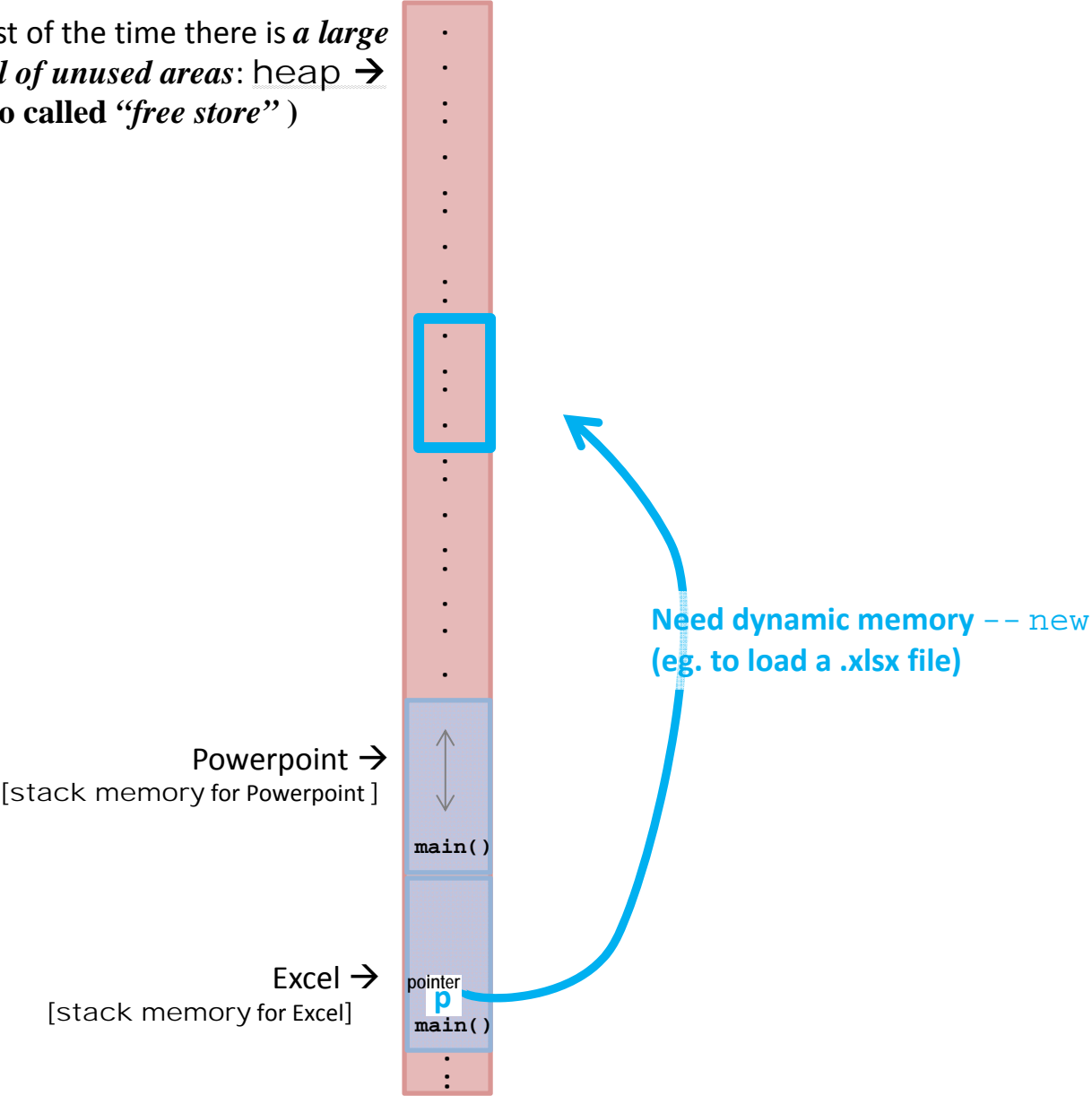
# Computer Memory

Most of the time there is *a large pool of unused areas*: heap → (also called “*free store*” )



# Computer Memory

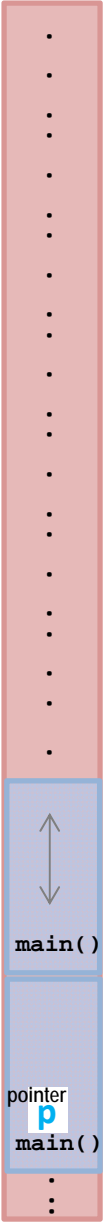
Most of the time there is *a large pool of unused areas*: heap → (also called “free store” )





# Computer Memory

Most of the time there is *a large pool of unused areas*: heap → (also called “*free store*” )



→ Space is returned to the heap -- delete

Powerpoint →  
[stack memory for Powerpoint]

Excel →  
[stack memory for Excel]

main()

pointer  
p  
main()

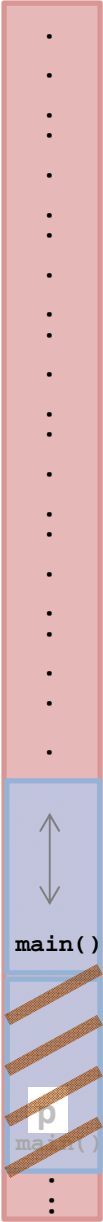
Illustration 1 [ Excel and Powerpoint ]

# Computer Memory

Most of the time there is a *large pool of unused areas*: heap → (also called “*free store*” )

Powerpoint →  
[stack memory for Powerpoint ]

Excel →  
[stack memory for Excel]

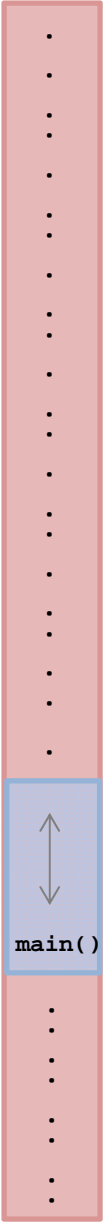


→ Space is returned to the heap -- delete

When excel is closed, the space becomes vacant  
→ Stack space is returned to the heap – excel.exe is ended

# Computer Memory

Most of the time there is *a large pool of unused areas*: heap → (also called “*free store*” )



→ Space is returned to the heap -- delete

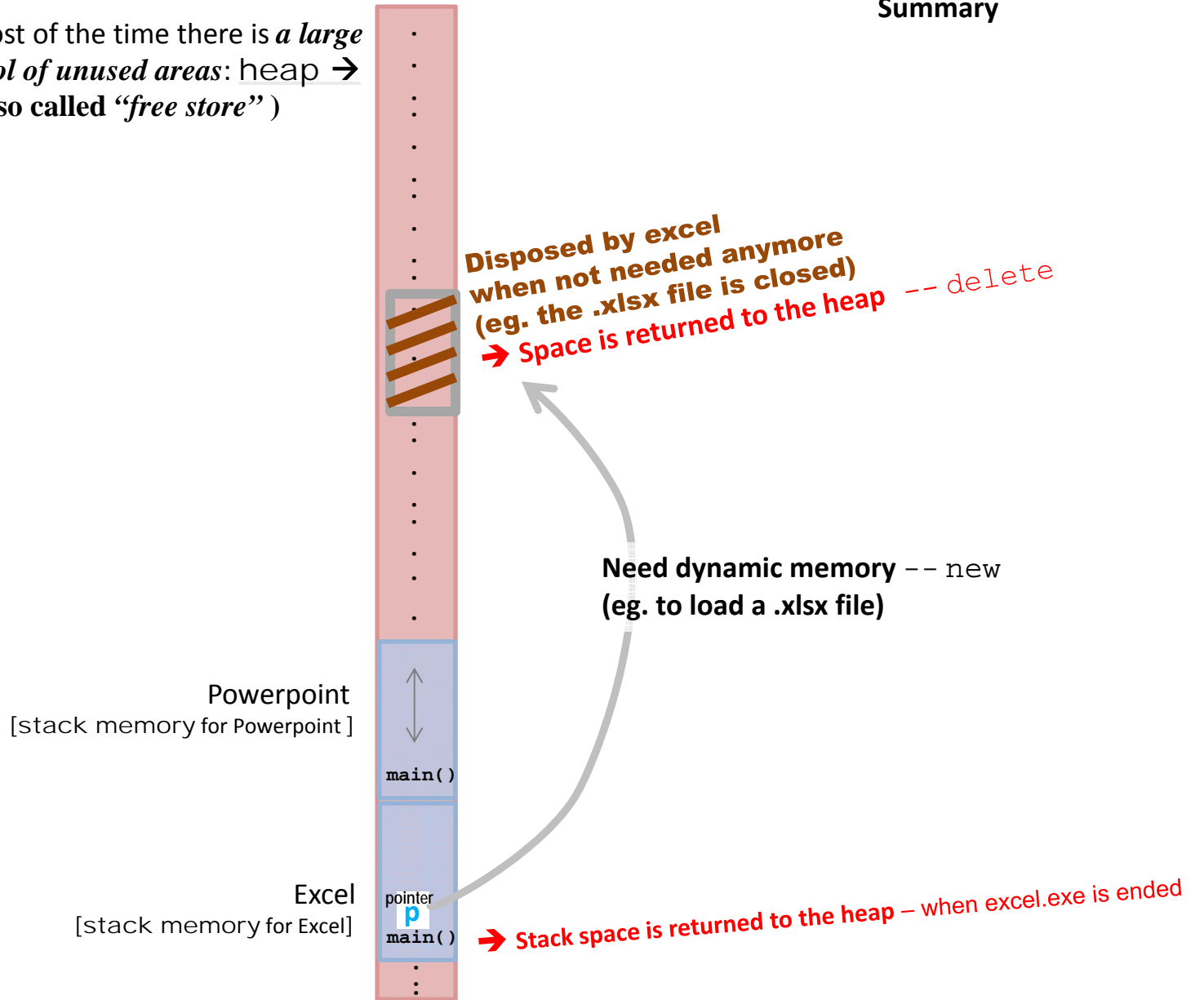
Powerpoint →  
[stack memory for Powerpoint ]

→ Stack space is returned to the heap – excel.exe is ended

# Computer Memory

Most of the time there is *a large pool of unused areas*: heap → (also called “free store”)

Illustration 1 [ Excel and Powerpoint ]  
Summary



**Illustration 1**  
Excel and Powerpoint

**Illustration 2**  
Java JVM

# Computer Memory

Illustration 2 [ Java JVM ]

1. Most of the time there is *a large pool of unused areas:*

2. When **JVM** (Java Virtual Machine ) starts, it occupies a large part.

Two special partitions for running user programs:

2.1 **Stack** –running methods (their local variables etc..)

2.2 **Heap** – runtime data for all objects and arrays, Interact with GC (Garbage collector)

