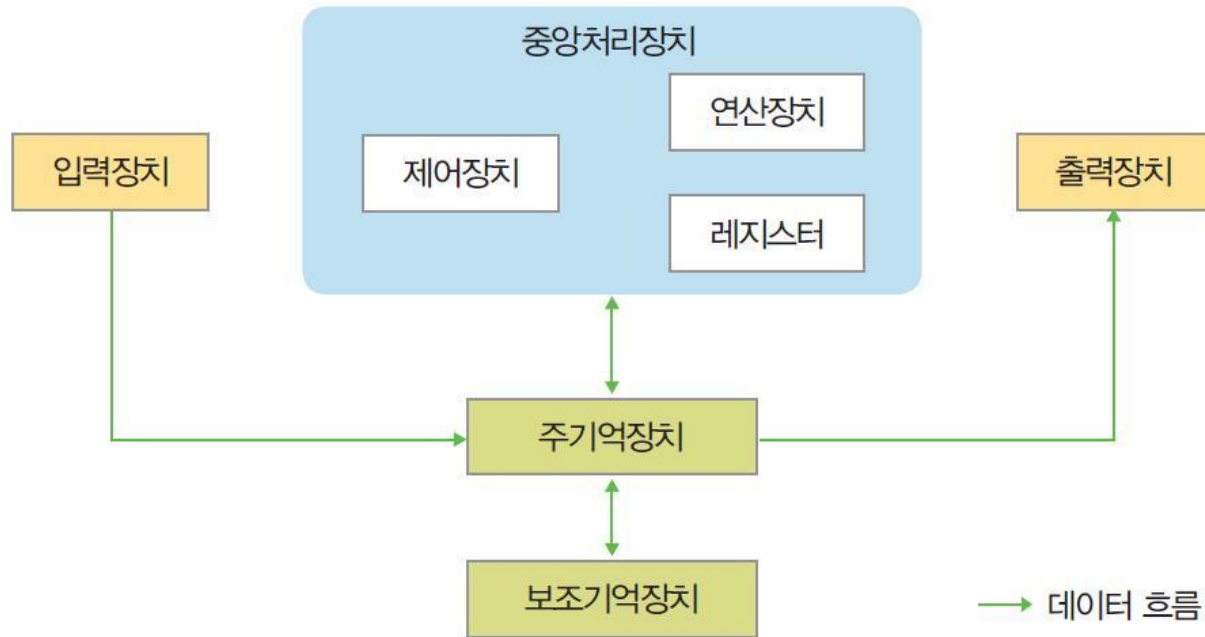


메모리 관리

- 메모리가 부족하면 컴퓨터 속도가 느려지는 이유.
- 메모리의 구조와 작동원리
- 메모리 릭을 해결하는 방법들

메모리가 부족하면 컴퓨터가 느려지는 이유



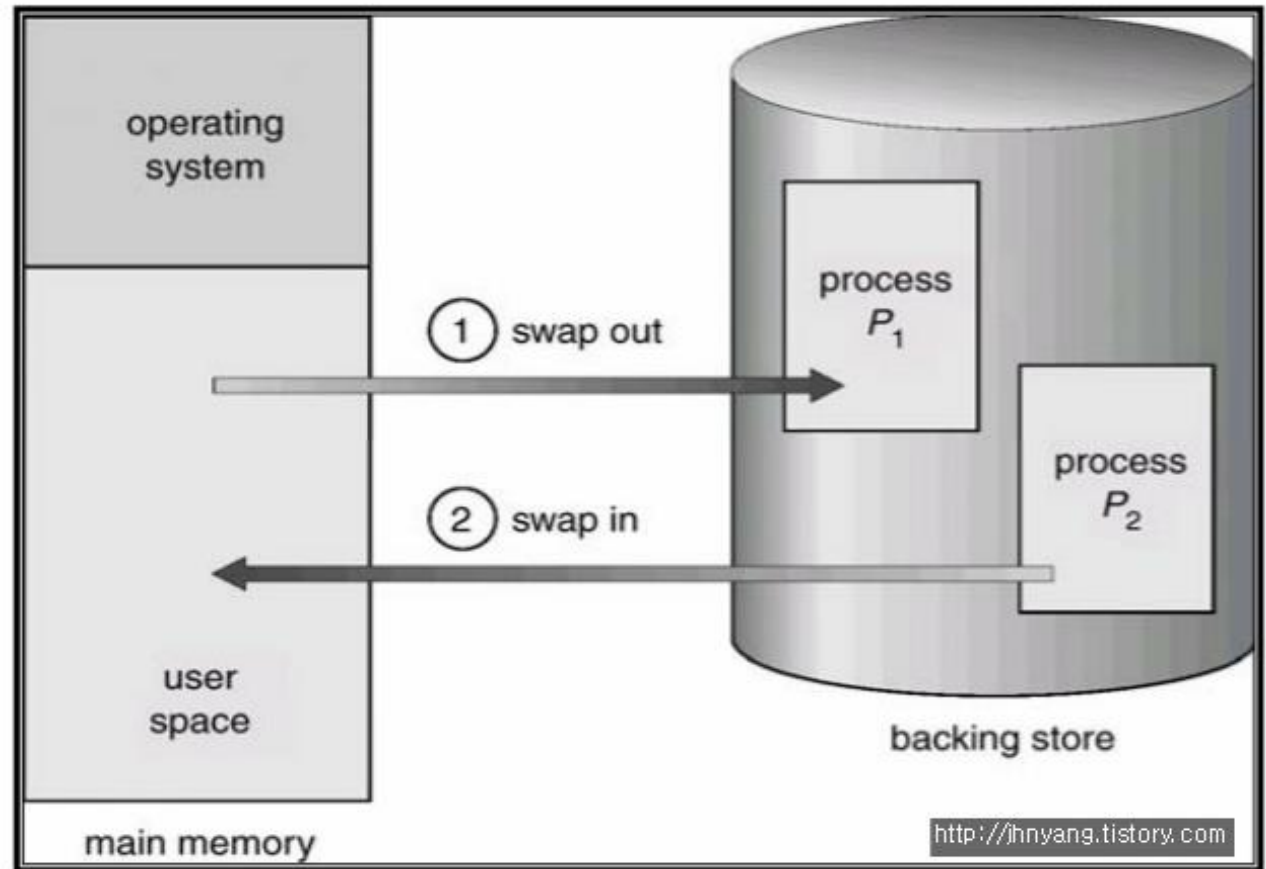
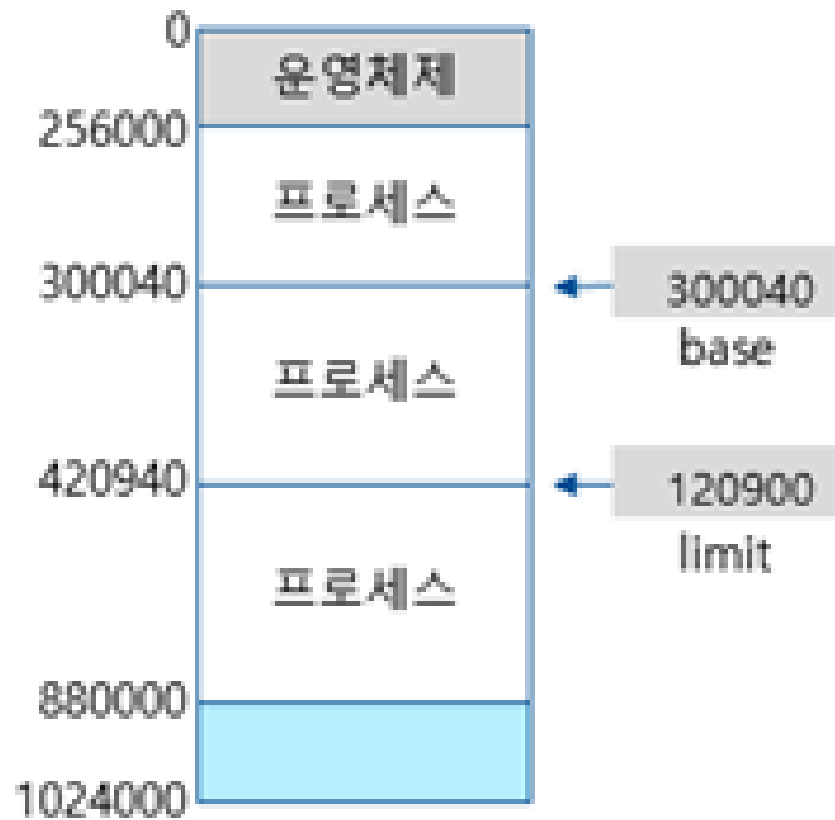
1. 프로세스(실행 중인 프로그램)가 저장되어 있는 공간은?
2. 메모리가 다 찼는데 새로운 프로그램을 실행하면 어떻게 될까요?
3. SSD를 달면 빨라지는 이유?

참조: <https://brunch.co.kr/@insuk/4>

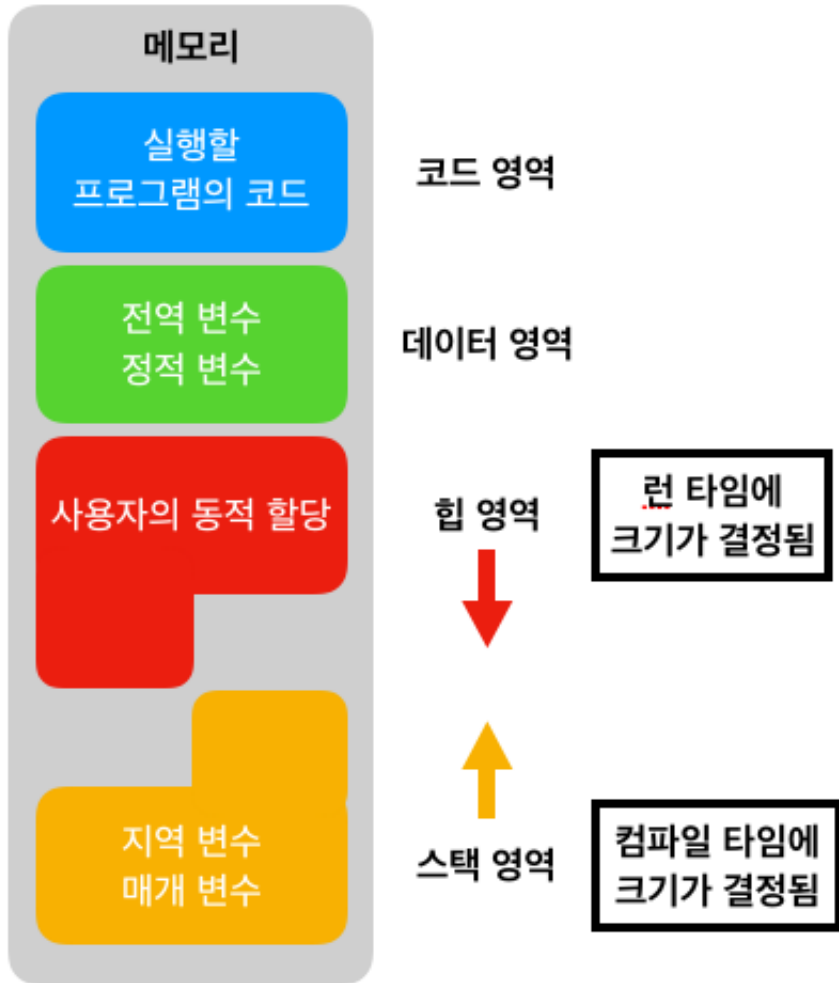
참고: <https://jhnyang.tistory.com/103>

참고: <https://infosense.tistory.com/24>

Swap Out & Swap In



메모리 구조와 작동원리



1. 코드영역과 이외의 영역이 구분되는 기준
2. 데이터 영역과 이외의 영역이 구분되는 기준
3. 힙영역과 스택영역이 구분되는 기준
4. 스택 영역은 왜 스택 자료구조를 사용했을 까요?
5. 참조변수(ex: String name)를 선언하고 객체를 생성. 이것은 힙과 스택 중 어디에 저장될까요?

참조: https://www.youtube.com/watch?v=vCKJacN_C3c

참조: <https://bskyvision.com/160>

스택영역

```
memory.cpp  X
(전역 범위)
1: #include <iostream>
2:
3: void fct1(int);
4: void fct2(int);
5:
6:
7: int k = 10;           → 전역변수
8: int h = 20;           → 전역변수
9:
10: int main(void)
11: {
12:     int m = 500;       → 지역변수
13:
14:     fct1(m);
15:     fct2(m);
16:
17:     int* s;
18:     s = (int*)malloc(sizeof(int));
19:
20:     *s = 211;
21:
22:     free(s);
23:
24:     return 0;
25: }
26:
27: void fct1(int a)       → 매개변수
28: {
29:     int aa = 1;        → 지역변수
30: }
31: void fct2(int b)       → 매개변수
32: {
33:     int bb = 2;        → 지역변수
34: }
```

bb = 2

b = 500

aa = 1

a = 500

m = 500

힙 영역

```
1 package TestJava;
2
3 public class TestJava {
4     public static void main(String[] args) {
5
6         String name = "chun";
7         String address = "부산";
8         if(true) {
9             String hobby = "독서";
10        }
11
12        name += "suyoung";
13
14    }
15 }
16
```

hobby = 주소300

address = 주소200

name = 주소100

스택영역

"chun"

"부산"

"독서"

"chun"

"suyoung"

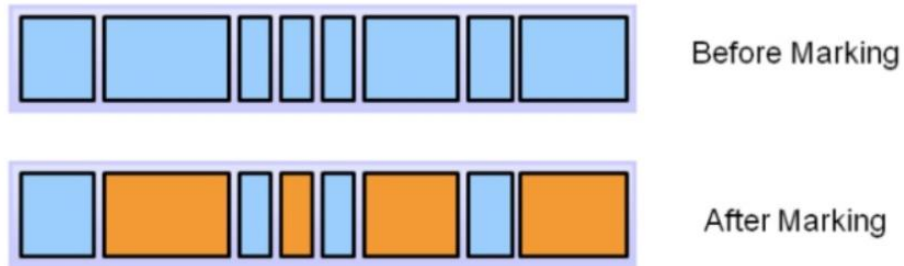
힙영역

메모리 릭을 해결하는 방법들

- Java > Garbage Collection
- Delphi > Free , Nil , Free and Nil

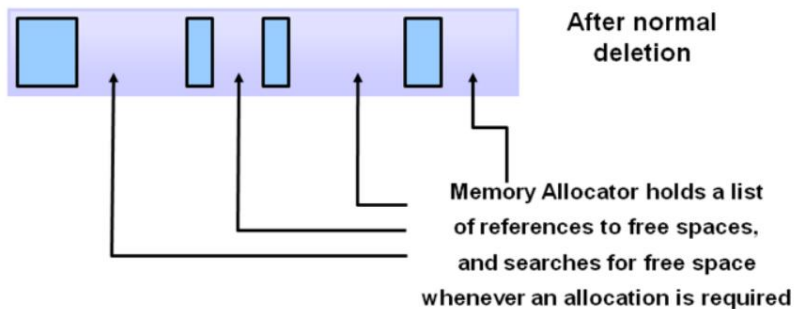
Java - Garbage Collection

Step 1 : Marking

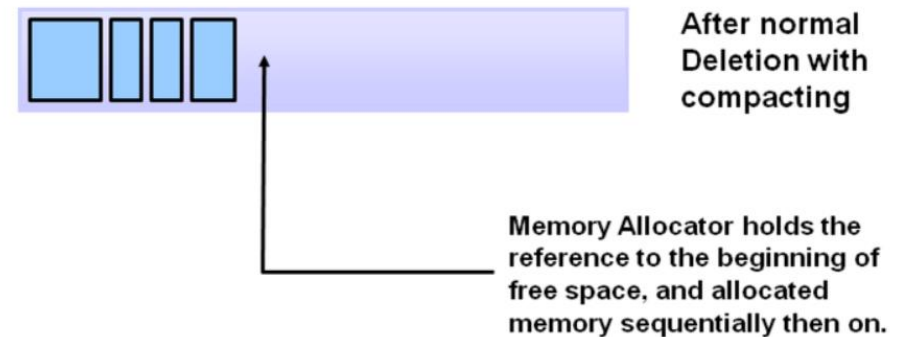


참고: <https://itmining.tistory.com/24>

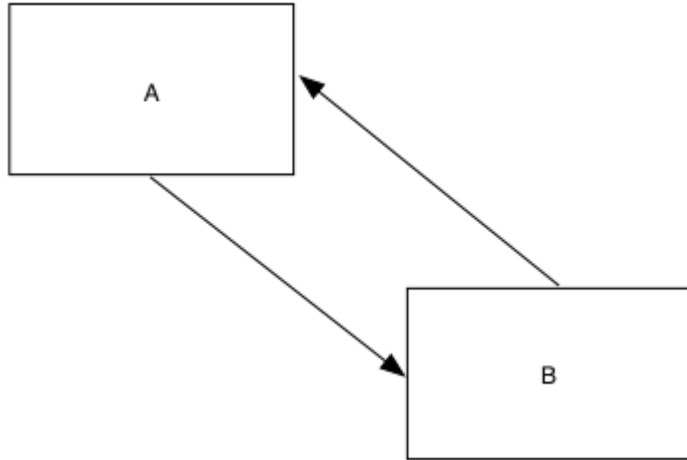
Step 2 : Normal Deletion



Step 3: Compacting



순환 참조



```
public class Bolt {  
    Nut nut = new Nut();  
}  
  
public class Nut {  
    Bolt bolt = new Bolt();  
}
```

Delphi – Free , Nil , Free and Nil

스택: record

힙 : class , dictionary

```
0012F59E 44 00 BC F5 12 00 F8 7A D.¼ö...øz
0012F5A6 42 00 F4 39 C0 00 00 00 B.ô9À...
0012F5AE 00 00 D5 0F AA 26 00 00 ..Ö.ª&..
0012F5B6 05 A1 02 00 00 01 FC F6 .;....üö
0012F5BE 12 00 C6 F3 42 00 F4 39 ..ÆóB.ô9
0012F5C6 C0 00 19 7B 42 00 38 F7 À...{B.8÷
```

할당 함수: Malloc, Calloc, New, Create

해제 함수: Free 메모리 해제 , Nil 포인터 해제 , FreeAndNil

참조:

https://tech.devgear.co.kr/index.php?mid=delphi_news&search_keyword=%EB%A9%94%EB%AA%A8%EB%A6%AC+%EA%B4%80%EB%A6%AC&search_target=title_content&document_srl=462375

참조: <https://m.blog.naver.com/PostView.nhn?blogId=an4tan&logNo=120021525180&proxyReferer=https:%2F%2Fwww.google.com%2F>