C언어 스터디 7차시

컴퓨터학부 23 유상원

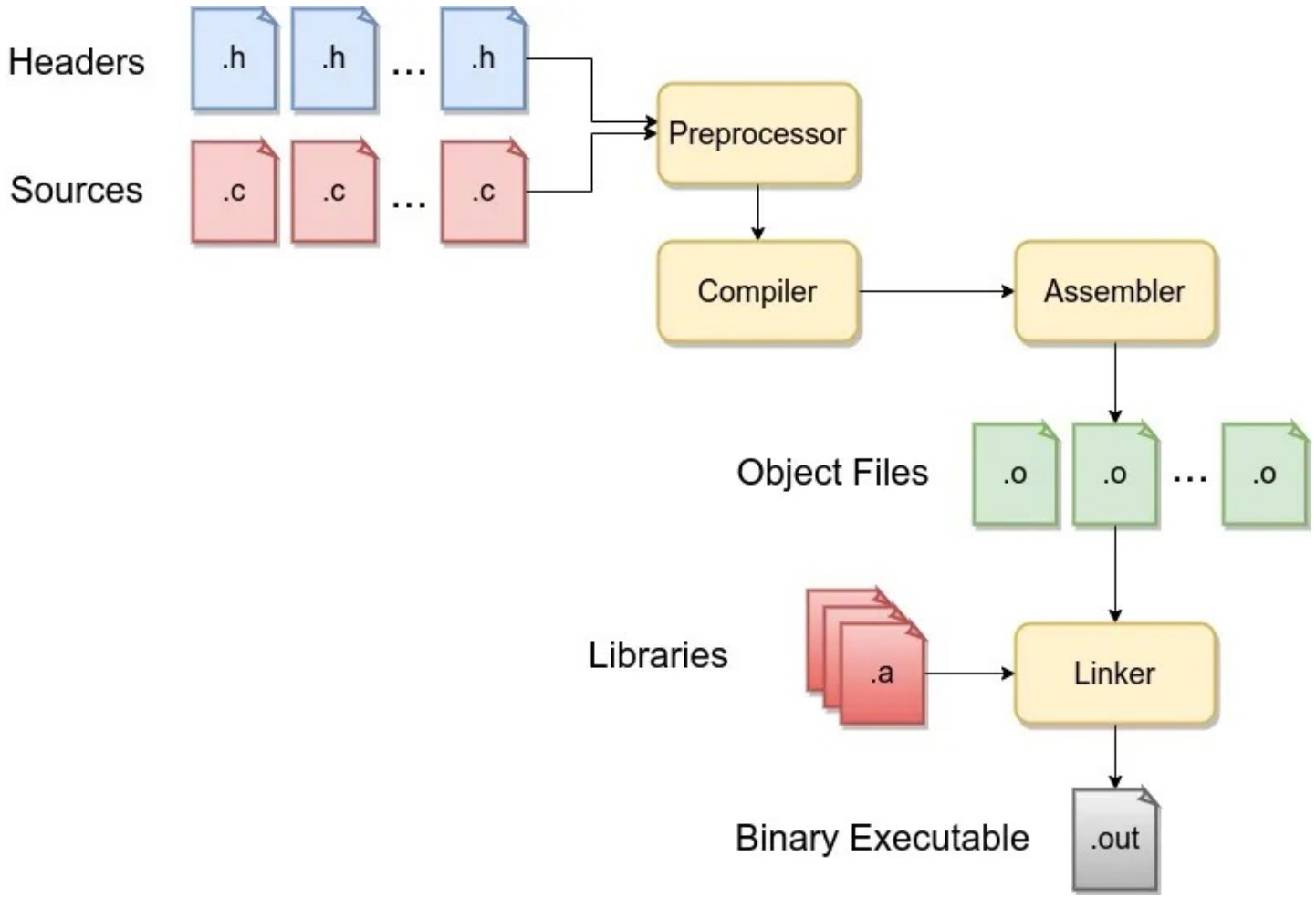
목차

- C언어 상식
  - 컴파일 과정
  - 메모리 구조
  - 캐시 히트

- C++ 맛보기
  - 자료구조: vector, queue, deque, priority_queue, map, set
  - 함수: sort, swap, min, max, upper_bound, lower_bound

컴파일 과정

컴파일 과정

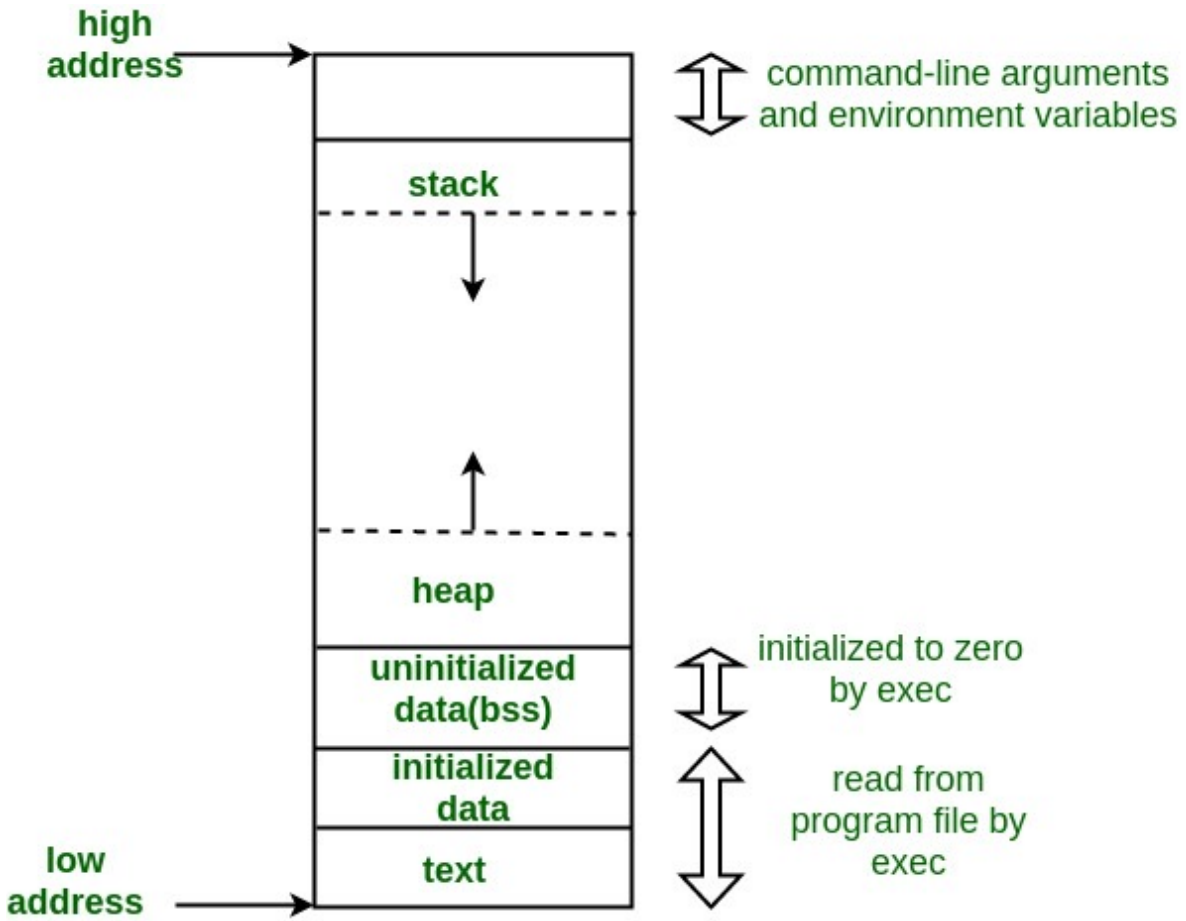


이미지 출처: https://www.stackzero.net/compilation-process-in-c/

컴파일 과정

메모리 구조

메모리 구조



이미지 출처: https://www.geeksforgeeks.org/memory-layout-of-c-program/

메모리 구조

캐시 히트

캐시 히트

- 둘 중 어느 코드가 더 빠르게 실행될까요?

```
int arr[10000][10000], sum = 0;

for(int i=0; i<10000; i++) {
    for(int j=0; j<10000; j++) {
        sum += arr[i][j];
    }
}
```

```
int arr[10000][10000], sum = 0;

for(int i=0; i<10000; i++) {
    for(int j=0; j<10000; j++) {
        sum += arr[j][i];
    }
}
```

캐시 히트

- 레지스터 > (캐시 메모리) > 메인 메모리
  - 속도
  - 가격

- 레지스터 < (캐시 메모리) < 메인 메모리
  - 크기

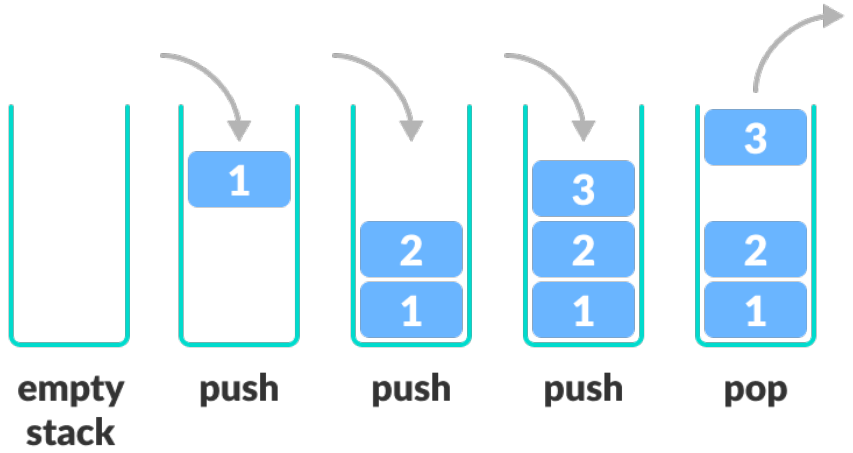
캐시 히트

- 메모리 지역성
  - 시간 지역성
  - 공간 지역성

C++ STL 자료구조

vector

- 원소가 연속적으로 저장되는 자료구조
- 배열처럼 인덱스로 접근할 수 있고, iterator를 사용할 수도 있음
- 사이즈가 고정된 배열과는 달리, 알아서 필요한 만큼 공간을 사용함
- 스택처럼 사용할 수 있음



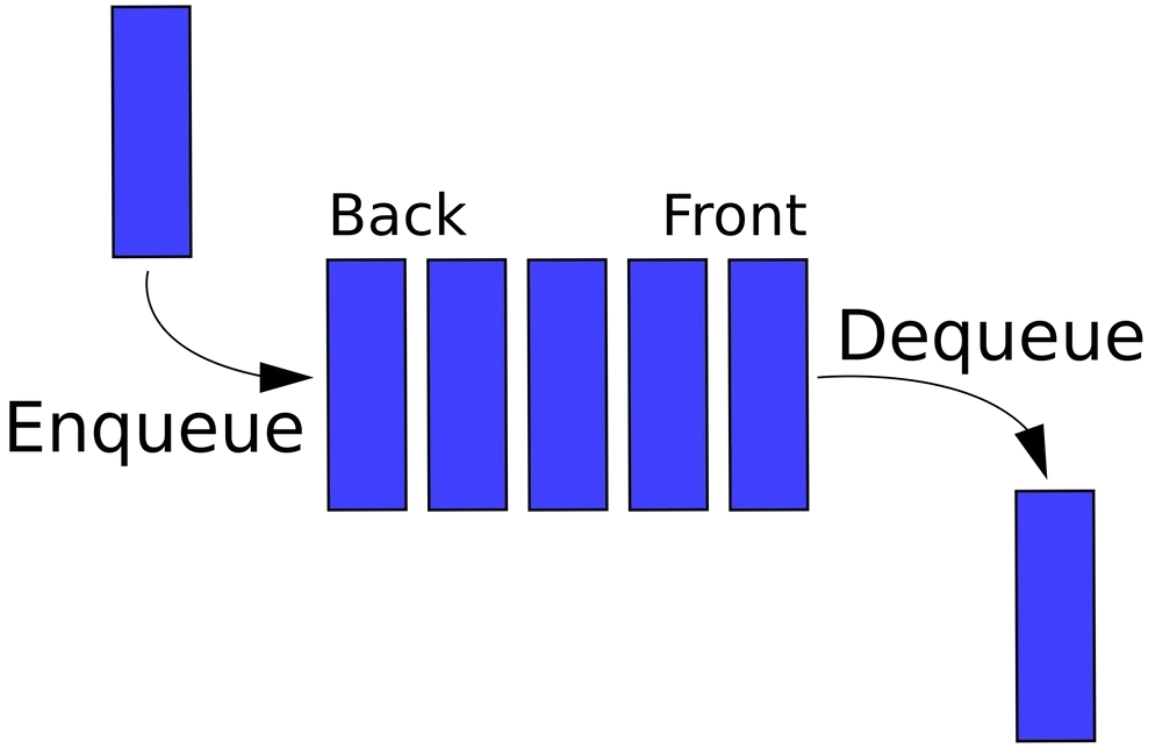
이미지 출처: https://hong-devbox.tistory.com/4

vector

- vector<int> v;
- v[idx];
- v.front();
- v.back();
- v.push_back();
- v.pop_back();
- v.size();
- v.empty();

queue

- 먼저 들어간 원소가 먼저 나오는 자료구조



이미지 출처: https://namu.wiki/w/큐(자료구조)

queue

- queue<int> q;
- q.front();
- q.back();
- q.push();
- q.pop();
- q.size();
- q.empty();

priority_queue

- 우선순위가 높은 원소가 먼저 나오는 자료구조
- heap 자료구조로 구현되어 있음

priority_queue

- priority_queue<int> pq;
- pq.top();
- pq.push();
- pq.pop();
- pq.size();
- pq.empty();

map

- key로 value에 접근할 수 있는, key-value 쌍을 저장하는 자료구조
- 내부적으로 key를 기준으로 정렬되어 저장되어 있음
- 주로 red-black 트리로 구현되어 있음

- unordered_map
  - ‘map’과 다르게 key를 기준으로 정렬되어 저장되지 않음
  - 주로 해시 테이블로 구현되어 있음

- multimap
  - ‘map’과 다르게 key가 중복될 수 있음

map

- map<int, int> m;
- m[key];
- m.find(key);
- m.size();
- m.empty();

set

- 집합, 원소의 중복을 허용하지 않음
- 내부적으로 key를 기준으로 정렬되어 있음
- 주로 red-black 트리로 구현되어 있음

- unordered_set
  - ‘set’과 다르게 key를 기준으로 정렬되어 저장되지 않음
  - 주로 해시 테이블로 구현되어 있음

- multiset
  - ‘set’과 다르게 key가 중복될 수 있음

set

- set<int> s;
- s.find(key);
- s.size();
- s.empty();

C++ STL 함수

sort

- 원소 정렬
- sort(시작, 끝);
- sort(시작, 끝, 비교);

swap

- 원소 서로 뒤바꾸기
- swap(a, b)

min

- 최솟값 반환
- min(a, b);
- min({ a, b, c, ... });

max

- 최댓값 반환
- man(a, b);
- man({ a, b, c, ... });

upper_bound

- 범위 내에서 특정 값보다 큰 첫 번째 원소 반환
- upper_bound(시작, 끝, 값);
- ***범위 내의 원소들이 정렬되어 있어야 함***

lower_bound

- 범위 내에서 특정 값과 크거나 같은 첫 번째 원소 반환
- lower_bound(시작, 끝, 값);
- ***범위 내의 원소들이 정렬되어 있어야 함***

STL

- 수박 겉핥기 식으로 가볍게 설명하고 넘어갔음
- 더 많은 정보는 cppreference.com 사이트를 읽어보는 것을 추천
- 각 자료구조나 함수의 구현, 시간 복잡도 등을 공부해 보는 것을 추천

과제

클래스2 – 에센셜

문제 번호	제목
2609	최대공약수와 최소공배수
11050	이항 계수 1
1181	단어 정렬
2751	수 정렬하기 2
10814	나이순 정렬

‘https://noj.am/(문제 번호)’ 로 문제 페이지를 바로 열 수 있음

감사합니다