12주차 결과보고서

전공: 컴퓨터공학과 학년: 2학년 학번: 20231632 이름: Jumagul Alua

1. 사용한 자료구조와 시간 및 공간 복잡도

'.maz' 파일의 각 줄을 'deque<string>'에 저장하였다. 나중에 쉽게 접근할 수있게 하고, 각 줄을 보관해주는 역할을 한다.

```
for (auto row : buffer.getLines()) {
    inpt.push_back(row);
}
return true;
```

실험 전 생각한 방법이랑 차이가 없다. 작성한 부분을 설명하자면, 사용자가 선택한 파일의 경로를 가져와서, 파일이 존재한다면, 'ofBuffer'를 사용해 파일을 읽어준다. 그리고 'buffer.getlines()'를 사용해 각 줄을 반복하며, deque<string>인 'inpt'에 저장한다.

다음, draw() 함수에는 'inpt'에 저장된 데이터를 사용하여 미로를 그리는 작업을 해주었다. 역시 이 부누에서도 예비보고서에 언급하던 방법이이랑 별 차이 없다. 미로의 높이를 inpt.size()으로 가져오고, 미로의 길이를 inpt[0] 부터 설정해준다. Cell에는 현재 셀의 문자를 넣어준다. 그리고 아래처럼 가로벽과 세로 벽을 적절한 위치에 선을 그려줬다.

```
int height = inpt.size();
    if (height == 0) {
        return;
    }
    int width = inpt[0].size();
    for (int i = 0; i < height; i++) {
        for (int j = 0; j < width; j++) {
            char cell = inpt[i][j];
            int drawX = j * 10;
            int drawY = i * 10;
            switch (cell) {
            case '-':
                ofDrawLine(drawX - 10, drawY, drawX + 10, drawY);
```

```
                break;
            case '|':
                ofDrawLine(drawX, drawY - 10, drawX, drawY + 10);
                break;
            }
        }
    }
```

readFile()함수의 시간 복잡도 및 공간 복잡도:

- 파일을 열고, 존재하는지 확인하는데 O(1)시간이 걸리며, 파일 크기를 'L'이라 했을때, 'OfBuffer'로 파일 읽기의 시간 복잡도가 O(L)가 된다. 'buffer.getlines()'는 파일의 줄 수를 'N'이라 하며, 모든 줄을 읽어들이는데 O(N)의 시간복잡도가 되고, 마지막으로 각 줄을 'deque'에 추가하는데도 시간복잡도가 O(N)이다. 따라서 총 시간복잡도는 O(L+N)이 된다.
- 총 공간 복잡도는: O(L+N*W)이다. 여기서 L은 파일 크기이고, N은 줄 수 그리고 W는 줄의 평균 길이가 된다.

draw()함수의 시간 복잡도 및 공간 복잡도:

- 미로의 높이와 너비 가져올 시간복잡도는 O(1)이며, for loop을 두개 썼으니까 O(N*W)이가 되고, 'ofDrawLine'을 호출하는데 O(1) 시간을 갖는다. 따라서 총 시간복잡도는 O(N*W)이 된다.(N은 줄 수, W는 줄의 길이)
- 미로 크를 저장하고, 미로의 각 줄을 저장할거라, 총 공간 복잡도는 O(1)이 된다.

2. 실험을통해습득한내용을한내용

- 이번 실험을 통해 deque에 대해 알게 되고 처음 써봤다. 이번 실습에는 'deque'를 사용해 '.maz'파일의 각 줄을 관리할 수 있었다. 'vector'와 유사하게 작동되지만, push_front 및 pop_front와 같은 메소드를 통해 앞쪽에서도 효율적으로 요소를 추가하고 제거할 수 있다는 것을 배웠다. 파일에서 읽은 각 줄을 deque에 추가하여 데이터를 저장하였고, 이

를 통해 미로를 쉽게 시각화할 수 있었다.

- 또한, ‘auto’ 자료형에 대해 알고 있었지만, 이번에 처음 써보고 연습 할 수 있었다. 파일 처리에서 auto 키워드를 사용하여 반복문 내에서 타입을 명시적으로 선언하지 않고, 컴파일러가 추론하게끔 했다. 이를 통해 파일에서 줄을 읽고 처리하는 코드를 간단하고 명확하게 만들 수 있었다.