

어드벤처디자인

(레벨 1)

2021년

김영학, 황준하, 김성영, 윤현주

이 자료는 금오공과대학교 컴퓨터공학과 2학년의 “어드벤처디자인” 과목에서
사용하기 위해 제작되었습니다.

차 례

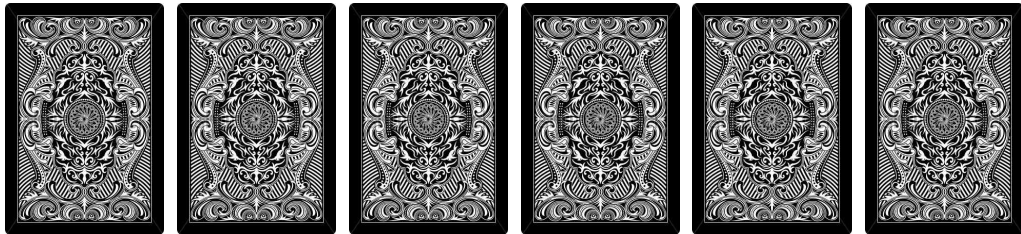
1. 레벨 1	1
1.1 숫자 카드 게임	1
1.2 항공사 고객의 우선순위 계산	3
1.3 술 취한 딱정벌레	4
1.4 칼라영상 반전 처리	5
1.5 간단한 메모리 관리자	7

1. 레벨 1

1.1 숫자 카드 게임 만들기

기본 문제

간단한 카드 게임을 제작한다. 게임 플레이어는 총 4장의 카드를 선택하며 4장의 카드 숫자의 합이 가장 큰 플레이어가 승리한다. 카드 숫자의 범위는 플레이어의 수에 의해 결정되는데 1부터 ($4 \times \text{플레이어의 수}$)가 된다. 예를 들어, 3명의 플레이어가 참여한다면 1부터 12사이의 숫자가 한 번씩만 나와야 한다.



제한요소 및 요구사항

- 플레이어 중 한 명은 컴퓨터가 되어야 한다(옵션 처리 가능).
- 게임이 시작되기 전에 카드는 미리 셔플링한다.
- 기본 메뉴를 제공한다.
- 상세한 입력과 출력 메시지를 제공한다.
- 게임은 대화식으로 진행한다(사용자가 차례대로 카드를 하나씩 선택한다).
- 최종 결과로 참여자들의 순위(sorting)와 점수를 출력한다.
- 2개 이상의 소스파일과 1개 이상의 헤더파일을 사용한다. (Java의 경우 2개 이상의 소스파일)
- 전역 변수를 사용할 수 없다.

확장 문제

카드의 숫자에는 동일한 숫자가 2개씩 존재한다. 동일한 숫자가 나온 경우에는 5배를 한다. 예를 들어, 1, 3, 4, 4인 경우 카드 숫자의 합은 $24(1+3+4*5)$ 이다. 추가적으로 조커를 사용할 수 있다.

1.2 항공사 고객의 우선순위 계산

기본 문제

한 항공사에는 이미 예약이 완료된 비행기에 대한 고객들의 후보 순위를 결정하기 위해 아래와 같은 식에 따라 대기 리스트(waiting list)를 운영한다.

$$\text{우선순위} = A - (B/B_MAX)*10 - (C/C_MAX)*5$$

A: 고객이 이미 예약 완료된 비행기를 예약하기 위해 접수한 순번

B: 고객이 과거 이용한 총 마일리지 점수, B_MAX: 최대 마일리지

C: 고객이 항공사의 해당 프로그램을 이용한 기간(일), C_MAX: 최대 기간(일)

이미 예약 완료된 비행기에 대한 대기 고객들에 대한 파일이 주어질 경우, 이 파일에서 개인별 정보를 입력하여 각 고객의 우선순위를 결정한 후 우선순위의 내림차순으로 정렬하여(우선순위가 높은 고객이 먼저 나옴) 화면에 출력하시오.

제한요소 및 요구사항

- 제공하는 고객 리스트 파일을 사용한다(열은 탭으로 분리).
- 정렬한 대기리스트는 텍스트 파일에 저장한다.
- 기본 메뉴를 제공한다.
- 상세한 입력과 출력 메시지를 제공한다.
- 2개 이상의 소스파일과 1개 이상의 헤더파일을 사용한다.
- 전역 변수를 사용할 수 없다.

확장 문제

빠른 검색을 위해 인덱스 트리(이진 탐색 트리나 해쉬 테이블)를 사용하여 고객을 관리한다(동일한 이름을 처리하기 위한 방법이 필요하다). 예약자의 검색, 추가, 삭제 및 갱신이 가능하며 갱신된 데이터를 파일에 주기적으로 저장한다.

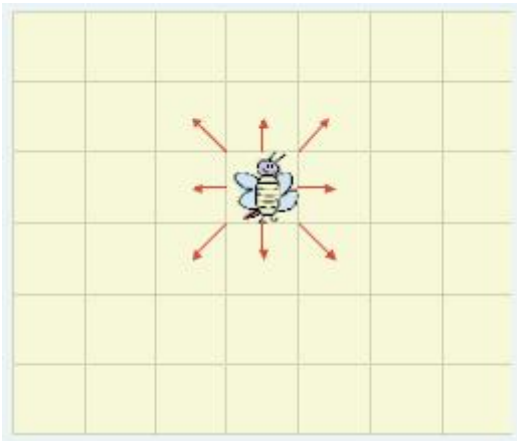
입력파일 형식

접수번호	이름	마일리지	가입년도
1	김연아	16436	2013-03-08
2	김재범	123462	2004-09-16
3	기성용	16374	1997-02-08
4	박주영	486	2000-05-19
.	.	.	.

1.3 술 취한 딱정벌레

개요

술 취한 딱정벌레가 같은 크기의 타일이 깔린 방 안에 있다. 이 벌레는 임의의 (random) 위치를 선택하여 여기저기 걸어 다니며, 현재 위치에서 주위의 8개 타일로 걸어 갈 확률은 동일하다고 가정한다. 딱정벌레가 그리는 경로는 어떤 모양일까? 그리고 딱정벌레가 방 안의 모든 타일을 한 번씩 지나가는데 걸리는 시간은 얼마일까?



이는 수학에서 “random walk”라 하는 문제인데, 랜덤 또는 랜덤에 가까운 움직임의 연속으로 이루어지는 작업, 예를 들어 기체 분자의 확산, 목초지를 찾아 헤매는 동물의 이동 경로, 주가 변동, 도박, 이동 통신에서 통신 노드의 이동 경로 등 다양한 분야에서 모델로 활용되고 있다. 정확한 값은 수학적으로 확률을 이용하여 풀어내어야 하지만 쉽지가 않다. 그러나 컴퓨터를 이용한 시뮬레이션을

을 하면 대략적인 추세를 볼 수 있어 아주 편리하고 유용하다.

기본 문제

딱정벌레가 방 안의 모든 타일을 한 번씩 지나가는데 걸리는 시간 및 통계자료를 구하시오.

- 1) 여러 번(10회 이상) 실행시켜 시행 별 소요 시간, 이동 횟수 및 통계(평균, 표준편차 등)값을 구하시오.
- 2) 방 크기를 변경하면서 실행하여 방 크기와 시간 사이의 관계를 예측해 보라. (방의 크기가 $10^3 \times 10^3$, $10^4 \times 10^4$, $10^5 \times 10^5$, $10^6 \times 10^6$, $10^8 \times 10^8$, $10^{10} \times 10^{10}$ 일 경우의 시간을 계산(예측)하시오. - 크기가 큰 방에 대해서 실행을 해 보면 아마도 시간이 너무 많이 걸려 최종 결과를 확인하지 못할 수 있음. 그러므로 “방 크기와 시간 사이의 관계를 예측해 보라”는 의미는 작은 크기의 방에 대한 실험 결과로부터 큰 크기의 방에 대한 결과를 예측해 보라는 의미임.)

제한요소 및 요구사항

- 결과 보고서 작성 시, 실험 결과(방 크기 및 딱정벌레 수에 따른 수행 시간 등) 및 방 크기 증가에 따른 예측 결과를 쉽게 이해할 수 있도록 표 또는 그래프 형태를 활용하여 표현한다. (엑셀 사용 권장)
- 기본 메뉴를 제공한다.

- 상세한 입력과 출력 메시지를 제공한다. 실행 결과에 대한 주요 정보만을 화면에 출력하고 자세한 내용은 파일에 저장한다.
- 2개 이상의 소스파일과 1개 이상의 헤더파일을 사용한다.
- 전역 변수를 사용할 수 없다.

확장 문제

딱정벌레의 마리 수에 따른 통계 결과와 그래프를 추가적으로 제시하시오.

참고 사항

보통 8개의 방향 중에 임의의 방향을 생성하려면 다음과 같이 생성하면 된다.

```
int direction = rand() % 8;
```

그런데, 이렇게 할 경우 매우 큰 문제가 생길 수 있다.

예를 들면, 0부터 7까지의 무작위 수가 골고루 등장하지 않는다든지... 등등.

이를 위해 다음과 같이 작성하면 이 문제가 어느 정도 완화될 수 있다.

```
int direction =(int) ((double) rand() / (RAND_MAX + 1) * 8);
```

// 0 이상 1 미만의 값으로 바꾼 후 8을 곱함

여기서 RAND_MAX는 rand() 함수를 통해 만들 수 있는 최대값(32767)으로 이미 표준 C/C++ 라이브러리에 정의되어 있다.

한편, 표준 C++에서는 mt19937이라는 새로운 random number generator를 만들어 더 균일하게 무작위 수를 생성할 수 있도록 만들었다.

사용 방법은 다음과 같다.

```
std::random_device rd;
```

```
std::mt19937 gen(rd()); // rd는 srand에서 초기화하는 역할
```

```
std::uniform_int_distribution<> dis(0, 7);
```

// 0부터 7까지의 값 중 하나를 생성함. <>는 디폴트로 <int>

```
int direction = dis(gen); // 0부터 7까지 값 중 하나가 나옴
```

1.4 칼라영상 반전 처리

개요

칼라영상을 저장하는 파일의 종류는 다양하다. 그 중에서 이진 PPM(raw Portable Pixel Map)은 매우 간단한 헤더를 갖는 구조로 칼라 영상을 저장한다. 3×2 크기를 갖는 파일의 구조는 다음과 같다. 처음 2바이트는 PPM 파일의 식별자(magic number) “P6”의 값을 갖는다. 16진수로는 0x5036이다. 다음은 영상의 가로 및 세로 길이이다. 마지막으로 최대 밝기 값이 나타난다. 이 값들은 하나 이상의 공백 문자(blanks, TABs, CRs, LFs, VTs, FFs)로 구분된다. 이후에는 픽셀의 값이 나타나는데 칼라 영상에서는 픽셀은 기본 색인 R, G, B의 값으로 구성된다. 픽셀들은 연속적으로 저장된다. 최대 밝기는 1에서 65535 사이의 값을 가진다. 최대 밝기가 256 미만이면 각 픽셀의 기본 색은 1바이트, 256 이상이면 2바이트로 표현된다. 예를 들어, 다음의 영상은 아래와 같은 구조를 갖는다.



50	36	0A	3	0A	2	0A	32	35	35	0A	FF	00	00	00	FF	00	...
P	6		3		2		2	5	5		255	0	0	0	255	0	...

기본 문제

PPM 파일을 읽어 영상을 반전한 후에 그 결과를 새로운 PPM 파일에 저장하시오.

제한요소 및 요구사항

- 기본 메뉴를 제공한다.
- 상세한 입력과 출력 메시지를 제공한다.
- 2개 이상의 소스파일과 1개 이상의 헤더파일을 사용한다.
- 전역 변수를 사용할 수 없다.

확장 문제

영상을 반전한 후 주어진 위치에 사각형을 출력하고 그 결과를 새로운 PPM 파일에 저장하시오. 회색음영 영상을 저장하는 PGM 파일의 구조를 분석하여 파일 포맷에 따라 PPM 또는 PGM 파일을 처리할 수 있도록 하시오.

1.5 간단한 메모리 관리자

개요

C언어의 malloc() 함수는 사용자의 동적 메모리 할당 요청에 대해 사용 가능한 메모리가 있으면 요청한 크기만큼 할당하여 그 시작 주소를 반환한다. 이 과정은 운영체제에 의해 처리된다. malloc() 함수를 호출하면 이 요청은 운영체제에게 전달되는데 운영체제는 메모리 관리 절차에 의해 사용가능한 메모리 영역에서 요청한 크기만큼 응용프로그램에서 사용할 수 있도록 작업한 후에 시작 주소를 반환한다.

기본 문제

운영체제가 수행하는 작업과 비슷하게 작동하는 메모리 관리자 프로그램을 구현하시오.

동작 설명

1) 관리할 최대 메모리 크기 설정: 바이트 단위로 크기를 할당한다.

2) 메모리 관리 함수: myalloc(), myfree()

int myalloc (int request_size) - request_size 크기만큼 메모리 요청이 들어온 경우로, 공간이 있는지 검사하여 덩어리를 잘라 내어 시작 주소를 반환한다.

void myfree (int start_address, int return_size) - start_address에서 시작하는 return_size 크기의 공간을 관리자에게 반환한다. 반환된 공간이 다른 공간과 연속되는 부분이면 합쳐서 큰 공간을 만든다.

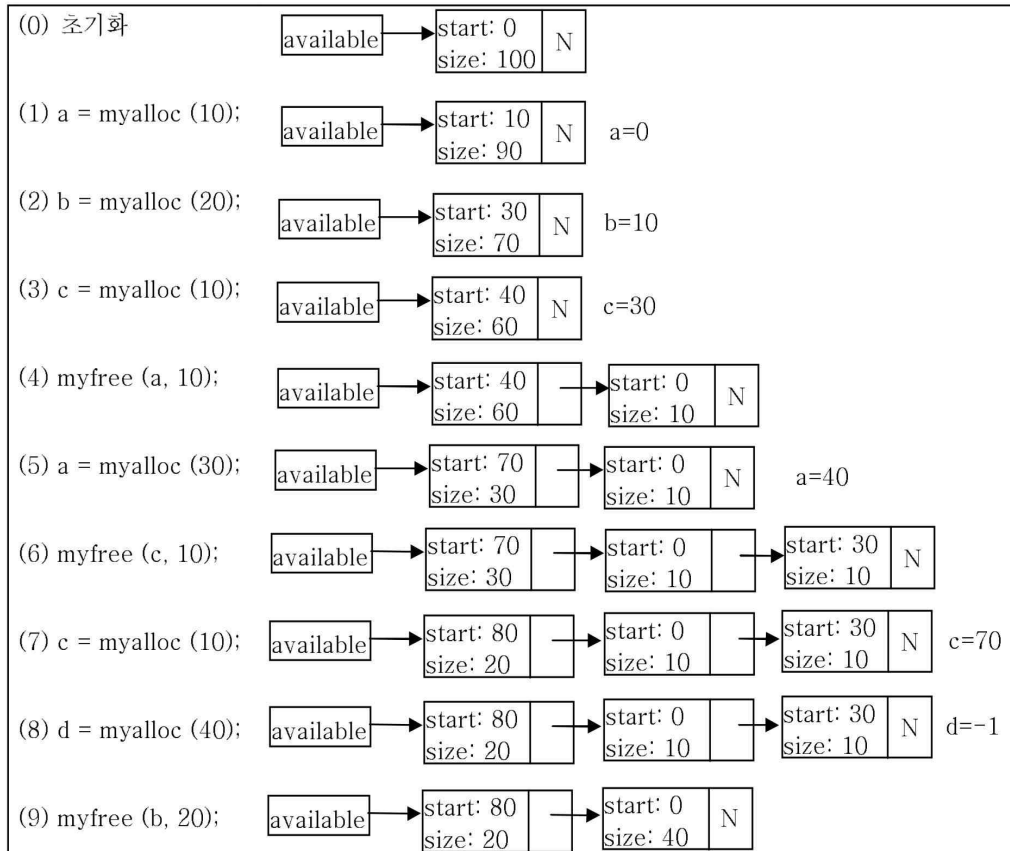
3) 가용 메모리 관리를 위한 자료 구조 (예시)

```
typedef struct chunk {  
    int start;           // 메모리 덩어리의 시작 주소  
    int size;           // 메모리 덩어리의 크기  
    struct chunk *link; // 다음 빈 공간에 대한 링크  
} Chunk;  
Chunk *available;
```

4) available 리스트는 할당하고 남은 덩어리(chunk)인 가용 공간에 대한 정보를 저장한다. 가용 공간은 크기와 위치가 제각각이고 할당과 반환이 반복됨에 따라 정보 내용이 변하므로 연결 리스트로 저장한다.

예를 들어 다음 그림과 같은 호출이 순차적으로 수행되었을 때 available 리스트의 변화를 보자. 전체 메모리의 크기는 100, 가용 공간의 앞부분에서부터 메

모리를 잘라 낸다고 가정한다.



※ 그림의 (8)번 단계의 요청에서 가용 공간은 40이 되지만 연속된 공간이 없으므로 -1을 반환하여 오류를 알린다. (9)번 단계에서는 반환된 공간이 0번지와 30번지에서 시작하는 공간과 연속되므로 합쳐서 크기 40의 가용 공간을 만든다.

제한요소 및 요구사항

- 기본 메뉴를 제공한다.
- 2개 이상의 소스파일과 1개 이상의 헤더파일을 사용한다.
- 전역 변수 사용은 가급적 자제한다.
- 세 가지 이상의 시나리오를 추가로 제작하시오. 즉, 예제의 코드 외에 메모리 할당/해제를 테스트해 보기 위한 코드를 작성하고 테스트해 보라.
- 자료구조의 설계는 임의로 변경할 수 있다.

확장 문제

- ① first-fit 외에 best-fit 및 worst-fit 방식으로 구현하여 그 결과 차이를 제시한다.
- ② 처리 과정의 중간에 강제적인 메모리 정렬 및 합병 기능을 제공한다.