꼬마 뱁새의 모험 기획서

SA 23기 프로그래밍 C언어 프로젝트

작성자 : 22기 임용식

**목차**

**1. 게임 개요**

1. 제목
2. 플랫폼
3. 장르
4. 표현 방식
5. 배경과 스토리
6. 기획 의도
7. 진행 방식 개요

**2. 게임 시스템**

1. 캐릭터 시스템
2. 아이템 및 재화 시스템
3. 인벤토리 시스템
4. 마을 시스템
5. 던전 시스템
6. 전투 시스템
7. 루팅 시스템

**3. 게임 컨텐츠**

1. 직업
2. 장비
3. 물약
4. 일반 공격
5. 스킬
6. 잡화점(마을 상점)
7. 여관(마을 상점)
8. 던전
9. 일반 몬스터
10. 보스 몬스터

**4. 프로그래밍 컨셉**

1. 랜덤 요소 구현 방식
2. 전체 UI 구현
3. 캐릭터 데이터 구현 방식 및 사용 공식
4. 캐릭터 생성 구현 방식
5. 장비 구현 방식
6. 인벤토리 구현 방식
7. 마을 구현 방식
8. 던전 구현 방식
9. 몬스터 및 보스 몬스터 구현 방식

1. 게임 개요

**1. 제목**

게임 제목은 “꼬마 뱁새의 모험” 입니다.

**2. 플랫폼**

PC게임이며, 콘솔 환경에서 구동됩니다.

**3. 장르**

텍스트 기반의 턴제 SUD(Single-User Dungeon)어드벤쳐 게임입니다.

**4. 표현 방식**

게임 내 모든 상황과 UI는 콘솔의 텍스트를 통해 구현합니다.

**5. 배경과 스토리**

둥지에서 통통한 배를 쓰다듬으며 엄마 뱁새가 주는 밥을 맛있게 먹는 낙으로 하루하루를 살아가던 꼬마 뱁새(플레이어)는 더 이상 둥지구석에서 밥만 축내지 말고 독립하라는 엄마 뱁새의 불호령에 결국 짐을 챙기고 둥지를 떠나 드넓은 세계를 향해 모험을 떠납니다…..

라는 스토리이지만 이 게임에서 스토리는 중요하지 않으니 자세히 신경쓰지 마십시오.

**6. 기획 의도**

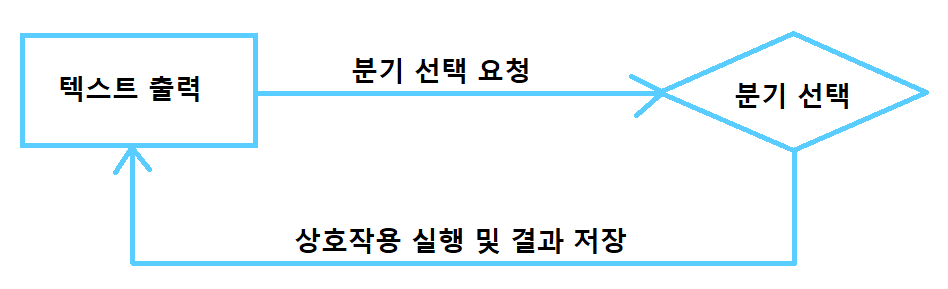
● 본 기획은 여러분들이 더 이상 연습문제만 풀다간 코딩에 지루함만 느끼고 동아리를 나갈까 봐 흥미를 당길 수 있을 만한 과제를 내 주기 위해 기획되었습니다.

● 또한 앞으로 동아리 활동을 하며 게임 제작 프로젝트를 할 여러분들에게 맛보기 수준의 게임 제작을 통해 타 부서(기획)와의 협업을 간접적으로 체험할 수 있도록 기획되었습니다.

**7. 진행 방식 개요**

본 게임은 게임 내에서 콘솔 창에 출력된 텍스트를 읽고 키보드를 통한 선택 메뉴 입력을 통해 상호작용을 진행해 나갑니다. 플레이어가 선택한 모든 상호작용의 결과는 텍스트를 통해 출력됩니다.

이를 flowchart로 표현하면 다음과 같습니다.



2. 게임 시스템

**1. 캐릭터 시스템**

캐릭터는 플레이어가 조종하는 대상입니다.

캐릭터는 다음과 같은 파라미터(능력치)를 보유합니다.

* 레벨

캐릭터의 레벨은 플레이어의 나머지 능력치 수준을 결정합니다.

캐릭터가 생성되는 시점에서 캐릭터의 레벨은 1이며, 전투를 통해 경험치 를 정해진 수치 이상 획득할 시 증가합니다.

* 공격력

공격력은 전투 이벤트에서 캐릭터가 몬스터에게 가하는 피해량을 결정합니다.

공격력은 레벨에 비례하여 결정되며 무기를 착용하여 추가로 증가시킬 수 있습니다.

* 방어력

방어력은 전투 이벤트에서 캐릭터가 몬스터로부터 입는 피해량을 결정합니다.

방어력은 레벨에 비례하여 결정되며 갑옷을 착용하여 추가로 증가시킬 수 있습니다.

* 체력

체력은 플레이어의 생존/사망 상태를 결정하는 능력치입니다.

체력이 0 이하가 되면 플레이어가 사망합니다.

체력은 레벨에 비례하여 결정됩니다.

**2. 아이템 및 깃털 시스템**

아이템은 플레이어가 소비하거나 장비할 수 있는 모든 것들을 총칭합니다.

아이템은 인벤토리에 저장하여 소지할 수 있으며,

깃털은 아이템을 구매하거나 마을에 존재하는 편의 시설을 이용할 때 사용할 수 있는 재화입니다.

깃털은 다음과 같은 방식을 통해 획득할 수 있습니다.

* 몬스터/보스 몬스터와의 전투에서 획득(루팅)
* 소지한 아이템의 판매

**3. 인벤토리 시스템**

인벤토리는 획득한 아이템을 저장할 수 있는 시스템입니다.

인벤토리는 최대 16칸의 저장 공간을 가지며, 모든 아이템은 개당 1칸의 저장 공간을 차지합니다.

인벤토리는 다음과 같은 기능을 포함합니다.

* 저장

구매 또는 획득한 아이템을 저장합니다. 인벤토리에 여유 공간이 존재하는 경우 자동으로 아이템이 인벤토리에 저장되지만, 공간이 가득찬 경우 버리기 기능을 실행할 지의 여부를 묻는 경고 메시지가 등장합니다.

* 버리기

저장 공간에 저장되어 있는 아이템을 삭제합니다.

* 장착

저장된 아이템 중에 장착 가능한 아이템을 선택해 장착합니다. 이미 장착한 아이템이 있는 경우 장비 공간에 위치한 아이템과 위치를 교환합니다.

**4. 마을 시스템**

마을은 여러 가지 편의시설이 모여 있는 공간입니다. 마을에서 이용할 수 있는 편의 시설은 다음과 같습니다.

* 잡화점
* 여관

**5. 던전 시스템**

던전은 몬스터들이 서식하는 공간입니다. 던전은 층별로 구성되어 있고 10의 배수층을 제외한 각 층에 진입할 때 마다 1회의 이벤트가 발생합니다.

발생 이벤트는 다음과 같습니다.

* 몬스터 발견
* 몬스터의 기습
* 보물 상자 발견

10의 배수층에는 다음 이벤트가 발생합니다.

* 보스 몬스터와 전투

발생한 이벤트가 종료되고 난 뒤 선택 이벤트가 등장합니다.

선택 이벤트는 다음과 같습니다.

* 다음 층으로 내려가기
* 아이템 사용하기
* 귀환(마을로 이동)

**6. 전투 시스템**

전투는 던전에서 (보스)몬스터와 마주쳤을 때 발생하는 시스템입니다. 모든 전투의 과정은 턴제 형식으로 진행되며, 자신의 턴 마다 행동 메뉴들 중 하나를 선택하여 행동이 가능합니다. 하나의 행동이 끝나면 자동으로 몬스터의 턴으로 턴이 넘어갑니다.

행동 메뉴는 다음과 같이 구성됩니다.

* 일반 공격

몬스터를 지정해 일반 공격을 시전합니다.

* 스킬

몬스터를 지정해 스킬 공격을 시전합니다.

* 아이템 사용

인벤토리에 저장된 소비 아이템을 사용합니다.

* 도망

일정 확률로 전투에서 이탈합니다. 전투에서 이탈에 성공할 시 마을로 귀환하며 이탈에 실패할 경우 아무런 행동도 하지 않고 몬스터의 턴으로 턴이 넘어갑니다.

**7. 루팅 시스템**

루팅 시스템은 전투에서 승리하거나 보물 상자를 열 경우 일정 확률로 아이템 또는 재화를 획득하는 시스템입니다.

일반 몬스터와의 전투를 통해 획득할 수 있는 아이템은 다음과 같습니다.

* 깃털
* 체력 물약
* 지혜의 물약

보스 몬스터와의 전투를 통해 획득할 수 있는 아이템은 다음과 같습니다.

* 깃털
* 체력 물약
* 지혜의 물약
* 무기
* 투구
* 갑옷

3. 게임 컨텐츠

**1. 직업**

게임 내에서 직업은 총 3가지가 존재하며, 다음과 같습니다.

* 전사 뱁새
* 마법사 뱁새
* 도적 뱁새

각 직업마다 능력치나 장비의 차이는 없으나 다른 스킬을 가지게 됩니다.

**전사 뱁새**는 강력한 단일 공격과 방어력 상승 버프 스킬을 특징으로 하는 직업입니다.

**마법사 뱁새**는 광역 공격 스킬을 특징으로 하는 직업입니다.

**도적 뱁새**는 다양한 특수 스킬을 특징으로 하는 직업입니다.

각 직업별 스킬은 스킬 컨텐츠 기획을 참고해 주세요.

**2. 장비**

장비는 캐릭터가 장착이 가능한 아이템들을 총칭합니다.

게임 내에서 장비는 총 3가지 카테고리가 존재하며, 다음과 같습니다.

* 무기

무기는 착용 시 플레이어의 공격력을 상승시키는 장비입니다.

무기는 총 5종류가 존재하며, 다음과 같습니다.

* 나무 부리
* 돌 부리
* 철 부리
* 황금 부리
* 초합금 부리
* 투구

투구는 착용 시 플레이어의 방어력을 상승시키는 장비입니다.

투구는 총 5종류가 존재하며, 다음과 같습니다.

* 나무 껍질
* 돌 껍질
* 철 껍질
* 황금 껍질
* 초합금 껍질
* 갑옷

갑옷은 착용 시 플레이어의 방어력을 상승시키는 장비입니다.

갑옷은 총 5종류가 존재하며, 다음과 같습니다.

* 나무 뱃살
* 돌 뱃살
* 철 뱃살
* 황금 뱃살
* 초합금 뱃살

**3. 물약**

물약은 소비 아이템으로 분류되며, 사용 시 플레이어의 체력을 상승시키거나 스킬 사용 횟수를 초기화할 수 있습니다.

게임 내에서 물약은 총 2가지가 존재하며, 다음과 같습니다.

* 체력 물약

캐릭터의 체력을 20 회복시킵니다.

* 지혜의 물약

캐릭터의 스킬 중 하나를 지정해 스킬 사용가능 횟수를 초기화합니다.

**4. 일반 공격**

일반 공격은 전투에서 (보스)몬스터를 대상으로 가능한 행동이며, 제한 횟수가 존재하지 않습니다.

**5. 스킬**

스킬 공격은 전투에서 (보스)몬스터를 대상으로 가능한 행동이며, 제한 횟수가 존재합니다.

제한 횟수는 던전 진입시 초기화되며, 지혜의 물약을 사용해 초기화할 수 있습니다.

각 직업별 스킬은 다음과 같습니다.

* 전사 뱁새
* 정수리 쪼기

한 마리의 몬스터를 지정하여 강력한 단일 공격을 시전합니다.

* 뱁뱁!

늠름한 울음소리로 3턴 간 방어력을 증가시킵니다.

* 연속 쪼기

한 마리의 몬스터를 지정하여 3회의 공격을 실시합니다.

* 마법사 뱁새
* 화염 깃털

모든 몬스터에게 광역 공격을 시전합니다.

* 번뜩이는 지식

이 스킬을 제외한 모든 스킬 사용 횟수를 초기화합니다.

* + - 운석 충돌

모든 몬스터에게 강력한 광역 공격을 시전합니다.

* 도적 뱁새
* 뒤통수 쪼기

한 마리의 몬스터를 지정하여 단일 공격을 시전하고 체력을 약간회복합니다.

* 비열한 날갯짓

100% 확률로 전투에서 도망칩니다.

* 물약인 줄 알았지?!

한 마리의 몬스터를 지정하여 20의 데미지를 가합니다. 체력 물약 1개가 소모됩니다.

**6. 던전**

던전은 진입 시 0층부터 시작하며, 0층에서는 아무런 이벤트도 발생하지 않습니다.

던전의 새로운 층에 진입했을 경우 발생하는 발생 이벤트의 각 발생 확률은 다음과 같습니다.

* 몬스터 발견(40%)

’몬스터 발견’ 이벤트 발생 시 1~3마리 중 임의의 몬스터 개체와 전투를 시작합니다.

몬스터 개체 수 결정 확률은 다음과 같습니다.

* 1마리(50%)
* 2마리(30%)
* 3마리(20%)

몬스터 발견 시 플레이어의 턴부터 시작하며, 플레이어가 도망에 성공하거나 모든 몬스터를 처치, 또는 사망하기 전까지 지속됩니다.

* 몬스터의 기습(30%)

‘몬스터의 기습’ 이벤트는 전투 시 몬스터의 턴부터 시작한다는 점을 제외하고 ‘몬스터 발견’ 이벤트와 동일하게 진행됩니다.

* 상자 발견(30%)

‘상자 발견’ 이벤트는 일정 확률로 아이템과 재화를 획득할 수 있는 이벤트입니다.

해당 이벤트를 통해 아이템과 재화를 획득할 확률은 다음과 같습니다.

* 골드 발견(20%)
* 체력 물약 발견(20%)
* 지혜의 물약 발견(20%)
* 무기 발견(3%)
* 갑옷 발견(3%)
* 투구 발견(3%)
* 아무것도 획득하지 못함(31%)

이 중 무기, 갑옷, 투구 발견 시 획득하는 장비는 던전 층 수마다 달라집니다.

**7. 여관(마을 상점)**

여관은 플레이어가 골드를 소모해서 잃은 체력을 전부 회복할 수 있는 편의시설입니다.

**8. 잡화점(마을 상점)**

잡화점은 플레이어가 깃털을 통해 아이템을 구매할 수 있는 편의시설입니다.

**9. 일반 몬스터**

던전에서 발견할 수 있는 일반 몬스터는 총 3종류가 존재하며, 다음과 같습니다.

* 애벌레
* 투구벌레
* 나방

각 몬스터의 공격력, 방어력, 체력 수치는 플레이어 레벨 및 층수에 비례합니다.

**10. 보스 몬스터**

보스 몬스터는 던전의 10의 배수층에서 발견되며, 다음과 같은 종류가 존재합니다.

* 장수풍뎅이(10층)
* 사마귀(20층)
* 대왕지네(30층)

각 보스 몬스터의 공격력, 방어력, 체력 수치는 플레이어 레벨 및 층수에 비례합니다.

4. 프로그래밍 컨셉

**1. 랜덤 요소 구현 방식**

게임 내 각종 확률성 이벤트(던전 이벤트, 전투 시스템 내 도망, 루팅 시스템 등)는 랜덤 함수(rand, srand)를 이용해 구현할 것을 권장합니다.

완벽한 난수를 생성하기 위해 시스템 시간을 srand의 seed값으로 부여합니다.

예를 들어, 던전 이벤트 내 상자 발견 이벤트의 각 아이템 별 확률 구현을 다음과 같이 할 수 있습니다.

|  |
| --- |
| srand(time(NULL));  ….  int prb = rand()%100;  if(prb>=0 || prb<20)  {  /\*골드 발견 코드\*/  }  else if(prb>=20 || prb<40)  {  /\*체력 물약 발견 코드\*/  }  else if(prb>=40 || prb>60)  {  /\*지혜의 물약 발견 코드\*/  }  else if(prb>=60 || prb<63)  {  /\*무기 발견 코드\*/  } else if(prb>=63 || prb<66)  {  /\*갑옷 발견 코드\*/  }  else if(prb>=66 || prb<69)  {  /\*투구 발견 코드\*/  }  else  {  /\*아무것도 획득하지 못함 코드\*/  } |

**2. 전체 UI 구현**

게임 내 모든 UI는 각 Scene별 UI에 맞게 구현합니다.

구현을 위해 gotoxy, system(“cls”)등의 콘솔 제어 함수를 사용합니다.

**3. 캐릭터 데이터 구현 방식 및 사용 공식**

게임 내 캐릭터에게 필요한 데이터는 다음과 같습니다.

* 레벨(정수)
* 경험치(정수)
* 플레이어 닉네임(문자열)
* 공격력(정수)
* 방어력(정수)
* 체력(정수)
* 장비(무기, 투구, 갑옷 - 구조체)
* 인벤토리(구조체)
* 소지 깃털(정수)
* 직업 코드(정수)
* 스킬(구조체 포인터)

다양한 데이터를 하나의 구조체로 묶어 구현합니다.

**4. 캐릭터 생성 구현 방식**

게임에서 플레이어는 총 3개의 캐릭터를 생성할 수 있습니다.

캐릭터 슬롯 구현을 위해 구조체 배열을 사용합니다.

캐릭터 생성 Scene을 제작하며, 해당 Scene에서 닉네임, 직업을 입력 받아 구조체에 할당합니다.

**5. 아이템 구현 방식**

각 아이템은 분류별로 열거체를 이용해 구현합니다.

장비의 경우 열거상수를 해당 장비의 스탯(공격력, 방어력)으로 설정합니다.

**6. 스킬 구현 방식**

각 직업별로 스킬들을 멤버로 하는 구조체를 생성한 후, 캐릭터 구조체 안의 스킬 구조체 포인터 멤버로 해당 직업에 대응되는 스킬 구조체를 가리키도록 설정합니다.

**6. 인벤토리 구현 방식**

인벤토리 자체는 배열을 통해 구현하는 방법과 링크드리스트를 통해 구현하는 방법이 있습니다. 배열을 통해 구현하는 경우, 인벤토리 자체는 단순히 배열 자료형만을 이용해 구현할 수 있으나, 아이템 습득 시 자동으로 비어있는 배열 공간 중 맨 앞에 데이터를 삽입할 수 있도록 인덱스 포인터를 이용합니다.

예를 들면 다음과 같습니다.

|  |
| --- |
| char inventory[50];  int ivp=0;  …  /\*아이템 습득 시\*/  inventory[ivp++] = item; |

이후, 임의의 아이템을 삭제할 시 배열 포인터를 해당 아이템 인덱스로 옮긴 후 다음과 같은 작업을 통해 아이템들의 배열 내 위치를 앞당겨주어야 합니다.

|  |
| --- |
| /\*8번 인덱스 아이템을 버리거나 판매했다고 가정\*/  ivp=8;  for(;inventory[ivp];ivp++)  {  inventory[ivp] = inventory[ivp++];  } |

링크드 리스트를 통해 구현하는 경우 배열을 통한 구현보다 구현 방법이 복잡하나, 아이템의 자동 습득과 삭제가 용이합니다.

링크드 리스트 구조의 삭제 함수는 삭제하고자 하는 아이템을 선택 한 경우 해당 아이템코드를 멤버로 가지는 노드의 전 노드까지 리스트를 탐색한 후, 탐색에 성공하면 삭제 할 노드를 임시 포인터 변수에 저장하고 그 이전 노드의 노드 포인터가 삭제할 노드의 다음 노드를 가리키게 한 후 임시 포인터 변수에 저장한 노드를 free를 통해 메모리에서 해제시키는 식으로 구현할 수 있습니다.

**7. 마을 구현 방식**

마을 Scene을 구현합니다. Scene의 자세한 구조 및 메뉴는 세부 기획을 참고합니다.

**8. 던전 구현 방식**

던전 Scene을 구현합니다. Scene의 자세한 구조 및 메뉴는 세부 기획을 참고합니다.

**9. 몬스터 및 보스 몬스터 구현 방식**

몬스터와 보스 몬스터도 캐릭터와 마찬가지로 구조체를 이용해 구현합니다.

필요 구조체 멤버는 다음과 같습니다.

* 레벨
* 이름(몬스터 종류)
* 공격력
* 방어력
* 체력

몬스터와 보스 몬스터의 경우, 몬스터 생성 함수를 이용해 전투에 진입할 시 생성하고, 전투가 종료되면 데이터를 삭제합니다. 이를 위해 동적 할당을 이용해 함수를 구현합니다.