```
MODULE
      ai

VAR
      State: {patrol, takeABreak, eatFood, dropSnack, shoot, dive, home);
      Action: {getFood, finishBreak, enemySightedInBreak, enemySighted,
noEnemySighted, grenadeSighted};
      Timer: 0..30;
      Killed: 0..20;

ASSIGN:
      init(State) := patrol;
      init(Timer) := 0;
      init(Killed) := 0;

    next(Timer) :=
    case
        Timer >= 0 & Timer <= 30 & State = patrol: Timer + 1;
        Timer > 0 & Timer <= 30 & Action = finishBreak: 0;
    esac;

    next(Killed) :=
    case
        Killed >= 0 & Killed <= 20 & State = shoot: Killed + 1;
    esac;

    next(State) :=
    case
        action = grenadeSighted: dive;
        State = dive & action = enemySighted: shoot;
        State = dive & action = noEnemySighted: patrol;
```

```
        State = patrol & action = enemySighted: shoot;
        State = patrol & action = noEnemySighted & Timer >= 0 & Timer <
30: patrol;
        State = patrol & action = noEnemySighted & Timer = 30: takeABreak;
        State = takeABreak & action = getFood: eatFood;
        State = takeABreak & action = enemySightedInBreak: shoot;
        State = eatFood & action = finishBreak: patrol;
        State = eatFood & action = enemySightedInBreak: dropSnack;
        State = dropSnack: shoot;
        Killed = 30: home
        State = home: home
        TRUE: {Patrol};
    esac;

    next(Action) :=
    case
        State = takeABreak & action = noEnemySighted: getFood;
        State = eatFood: finishBreak;
        TRUE: {enemySightedInBreak, enemySighted, noEnemySighted,
grenadeSighted};
    esac;
```

Sample Solution for Assignment 2:

Requirement Generation Solution:

1. State = eatFood && action = enemySightedInBreak --> X(State = dropSnack)
2. State = shoot --> F(State = home)
3. State != takeABreak U(State = patrol)
4. EG(F(State != dive))
5. AG(Action = finishBreak --> X(State = patrol))