EFREI L1 Int.
Monday, February 26, 2018

Programming Techniques and Tools
Project 3 (V_0)

1 Project Presentation

The Kapture is a board game opposing two camps composed of several pawns with different roles. Each side has a flag that the opposing side has to capture and bring back to his own flag. The game is played on a board representing a field that may have various obstacles (forest, rivers). At the start of the game, each player occupies one side of the field and his pawns are grouped around its flag. The position of the flag may be chosen at random or provided by the player. The entire field is not visible, it must be explored gradually to discover the obstacles and monitor in constantly the movements of the opponent.

1.1 Game turn description

Each player performs the following actions on his/her turn

1. Move all his pawns according to the number of movement points and the field (see below) ;
2. Resolve battles when two enemy pawns are side by side at the end of movements ;
3. Test end of game, if the opponent's flag is brought to its own camp's flag.

1.2 Camp composition

Each camp is composed of several pawns, divided into several categories :

1. The Scouts (that can be materialized by an 'S') ;
2. The Infantryman (that can be materialized by a 'I', or 'C' like cannon fodder) ;
3. Shock Troops ('T').

Each of these categories has different characteristics for movement and for battles.
Move Points by Game Turn and Category :

Flag	0
Scout	5
Infantryman	3
Shock Troops	2

1.3 Movements

At the beginning of each turn, each pawn has all its points movement, which he may use in whole or in part, but points not used in a turn can not be accumulated for the next turn. Movements are made indifferently on one of the 8 squares adjacent to the current position of the pawn. Each move consumes a certain number of move points depending on the field (see bellow).

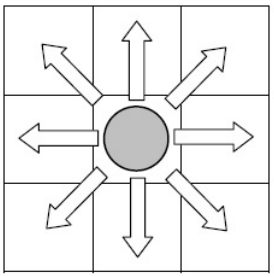

FIGURE 1 – Allowed moves for a pawn

A pawn can not move to a square occupied by another pawn of any kind.

When moving a pawn, a player may decide not to move the pawn while he still has movement points. The movement of this pawn is then completed for this turn, it is not possible to move it again in the same turn.

1.4 Influence of field type on movement

The number of movement points required for a move depends on the type of the **starting** position of the movement :

Field Type	number of points
Normal	1
Forest	2
River	all

Thus, when a pawn leaves a 'river' position, it consumes all its movement points to make a move, regardless of its category and regardless of the type of the arrival field. In addition, a move from a 'River' position can only be made if the pawn has all his movement points.

1.5 Catching Flag

Any pawn who is not a scout can catch the enemy flag by positioning himself on a box adjacent to the flag. In this case, he is considered as a scout for battles, but keep the moving points of his category. If the pawn bearing a flag loses a battle, the flag he wore falls randomly on a square adjacent to the box where this pawn was, the box where the flag falls must be empty. It can then be picked up by any pawn, which then becomes flag bearer.

1.6 Game over

A camp wins when both of the following conditions are met :

1. The flag-bearing pawn is on an adjacent square (even diagonally) to the initial position of the flag of his camp
2. The flag of his camp is at its initial position.

1.7 Resolving battles

Battles are not deadly ! Instead of being purely and simply withdrawn from playground, a pawn losing a battle is returned to its starting position, next to the initial position of his flag.

1. Scout vs. Scout : nothing happens ;
2. Infantryman or troops of shock against scout : the scout is returned to his starting position ;
3. Shock Troops against Infantryman : Infantryman is returned to his starting position ;
4. Infantryman against Infantryman : one of them (randomly chosen with 50% of chance) is returned to its starting position ;
5. Shock Troops Against Shock Troops : Both Shift One Space Towards their flag. If retreat is not possible (board edge or position occupied by other pawns) then return to the starting position.

1.8 Management of the war fog

Last subtlety of this game : the field is a priori unknown by both sides at the beginning of the game. It is during the different movements that the field is revealed (this is what we call the 'fog of war'). With each movement of a pawn, the 4 squares adjacent to the destination position of the movement are unveiled once and for all.

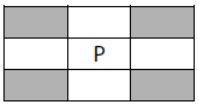

FIGURE 2 – Area of field discovery around a pawn P :

In addition, enemy units may not be visible on the unveiled map, but only if they are in the neighborhood V of a pawn :

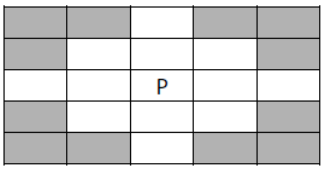

FIGURE 3 – Visibility zone V of a pawn P : boxes marked in white :

Thus, enemy pawns may move into an open area of of field discovery without being seen if no pawn is there to monitor what is happening.

The following points are left to your initiative :

— The size of the field

— The number of pawns in each camp ;

— The number of pawns in each category in each camp ;
— The determination of the starting positions.

Some options to dig, once the basic version of the project is finished :

— The development of a system allowing the game not to be blocked if a camp remains grouped around his flag ;
— The programming of an artificial intelligence ;
— The introduction of new types of areas (impassable buildings) ;
— The introduction of new types of pawns (a game optional : one must always be able to play the version containing only the three categories of pawn described in the subject) ;
— The presence of more than two camps on the ground ;
— And others according to your imagination and the progress of the project !

2 Schedule

The schedule of the project is as follows :

— February 26 : the project paper is submitted on `Campus Efrei`
— Marsh 1st : Quick-Off Session. During this session, you analyze the project, divide the work to do between project members and start working on the project. If you have any question, you can ask the teachers directly. **Don't wait until this date to discover the project subject** : come with concrete questions and clear ideas about the problems you have to face. At the end of this session, you **must** give the composition of the project teams to your teacher.
— Next, you will have three week time to work on the project at home. If you have any question, don't hesitate to send your question to your teachers.
— Marsh 22, 4 :00 pm : Project Defense. You have 20 minutes to present your project to the teachers. Further details about the precise schedule will be given later on.
— Marsh 22, 11 :55 pm : Project Submission. You must submit your project on Campus. You may fix bugs and/or shortcomings of your code after the defense, provided you list all the fixes in your report.

3 Deliverable

The deliverable consists of a single archive named `LASTNAME1.FirstName1.LASTNAME2.FirstName2.rar` or `.zip` to identify the project members. The archive contains :

— the report, named report.pdf
— the source files (`.c` and `.h` files only)
— a `readMe.txt` file containing a brief description of the realized part of the requirement, and the compilation/execution options.

The report is between 5 to 10 pages long. It includes an introduction, a short description of the project subject, some **algorithms** for the most important issues of the project, a discussion og the encountered problems/difficulties, and a conclusion. The report should not explain your code in details, let alone include portions of your code : good code should be self-explanatory. The code must be properly indented and commented. It must comply with the standard naming rules for variables and functions.

3.1 Minimal version

— No artificial intelligence

— DOS-like graphical interface with color character display (be careful, the realization of a more elaborate graphical interface is under your responsibility and can not compensate for the non-realization of another point of the subject !) ;
— Presence of help during the game on the different possibilities offered to the player ;
— Backup and restore of a game in progress (use files to save/restore the game state) ;

3.2 Programming constraints

— **No** global variables, goto, continue, break (except in a switch statement) ...
— Use functions
— Use modules (one `.c` and one `.h` file by theme)
— Use dynamic arrays to represent the field and to represent the set of pawns of each camp
— Comment your program

3.3 Project defense

Each project team has 20 minutes to defend its work, as follows :
— During the first 5 minutes, you will briefly introduce the project, then expose the results you achieved and the problems you faced. Both team members must speak in turn.
— For the next 15 minutes, you will run a demo of your program and show that you answered all (or a part of) the requirements. You have then to answer the questions of the teacher. Each team member must be able to answer any question, on any part of the project.

Please note that, depending on the quality of their answers, the project members may get different marks.

Finally, any detected plagiarism will be severely sanctioned.

Have fun