

Programmation C

THREE TO GO

Rapport de projet

Pierrejean BESNARD et Louis BILLAUT
TDA / TP1

Sommaire :

I. Présentation du projet.....	2
1. Origines.....	2
2. Objectifs.....	2
II. Planification du projet.....	3
1. Présentation des grands axes du projet.....	3
2. Répartition des tâches.....	4
III. Développement du projet.....	5
1. Descriptions des fonctions principales.....	5
2. Difficultés rencontrées.....	6

I. Présentation du projet

1. Origines

Le projet de programmation C du troisième semestre nous a été imposé par les enseignants, nous partions donc tous du même sujet avec une structure pré-codée dont nous devions nous servir.

Le sujet a suscité un intérêt particulier au sein du binôme, en effet, étant encore novices en programmation C, la mise en œuvre d'un programme aussi long et complexe dans ce langage s'annonçait comme un défi pour les deux membres du groupe.

Nous devions donc créer le jeu «Three To Go». Le principe du jeu était simple, des formes colorées appelées « token » devaient être générées aléatoirement et ensuite placées à droite ou à gauche d'une seconde liste, le but étant d'assembler des triplets de forme ou de couleur pour les faire se supprimer et ainsi augmenter le score du joueur. Le joueur devait pouvoir également décaler chaque formes de la même famille vers la gauche, ou chaque forme de la même couleur pour essayer d'assembler des triplets de couleur ou de forme. Le jeu devait être principalement utilisable via une interface graphique entièrement utilisable avec la souris, le tout à l'aide de la bibliothèque graphique de l'université.

2. Objectifs

L'objectif initial était donc de réussir à coder un algorithme capable de modéliser une partie de Three To Go classique, en affichant la liste des tokens générés aléatoirement en haut de la fenêtre de jeu, et la liste des tokens placés en bas. Il fallait également proposer une interface graphique

utilisable avec la souris permettant au joueur de cliquer sur deux boutons pour placer ses tokens (un pour les placer à droite et l'autre pour la gauche).

Chaque token du jeu était mis en forme grâce à la structure imposée par les professeurs, ainsi, ceux ci possédaient une forme et une couleur (toutes deux représentés par des entiers de 0 à 3).

La liste des tokens à placer générée aléatoirement était représentée par une liste de tokens chaînée simple, celle des tokens placés était quant à elle une liste chaînée circulaire, et doublement chaînée au niveau de ses attributs couleurs et forme.

L'habilité à manipuler les listes chaînées était donc clairement de mise, il fallait donc coder plusieurs fonctions capables de décaler, d'ajouter, de supprimer et de chaîner les tokens via leurs différents attributs.

Au-delà remplir les consignes dites «obligatoires» données dans le sujet, nous avons comme but de créer un véritable jeu à part entière, doté d'une interface graphique attrayante ayant un design travaillé comme un jeu pouvant être trouvé sur internet ou sur mobile. Le jeu devait pouvoir être agréable à jouer et être doté de différentes fonctionnalités comme un affichage de score, un chronomètre, et un menu d'accueil.

II. Planification du projet

1. Présentation des grands axes du projet

En se représentant une partie dite «type» du jeu Three To Go, nous pouvions discerner trois grands axes de travail:

- Modéliser les deux listes de tokens et les éléments liés aux actions possibles du joueur.
- Chaîner ou doublement chaîner les listes, simplement, par forme ou par couleur.

- Gérer les transferts de tokens d'une liste à une autre et les suppressions de triplets.
- Rendre possible le décalage par forme et couleur des tokens.

A ces trois grands axes venaient s'ajouter, trois tâches secondaires mais pas moins importantes pour le groupe à savoir:

- Mettre en place un chronomètre et un système de score.
- Inclure une mécanique de menus (écran d'accueil, écran de fin de partie).
- Travailler sur le visuel du jeu, avec des animations originales rendant le jeu plus attrayant.

2. Répartition des tâches

Pour une meilleure compréhension et une collaboration plus fructueuse, la plupart des fonctions liés au chaînage simple des formes, à la génération de tokens aléatoire, au déplacement de tokens d'une liste à l'autre et à la suppression des triplets ont été conçues à deux.

La partie mécanique des menus a été conçue par Pierrejean BESNARD qui a également conçu les thèmes visuels du jeu et a mis en place les différents boutons cliquables en jeu.

Louis BILLAUT quant à lui s'est occupée de la mise en place du chaînage par forme et couleur et aux fonctions de décalage des formes et couleurs ainsi qu'à la création du chronomètre.

III. Développement du projet

1. Descriptions des fonctions principales

Le programme THREE TO GO est disponible dans le même répertoire que celui de ce document. Le fonctionnement de la boucle principale est décrit dans celui-ci.

Les fonctions les plus importantes présentes dans le programme vont donc être décrites brièvement dans leurs fonctionnements.

token_aleatoire() Renvoie un pointeur sur un token de couleur et de forme aléatoire.

insere_simple() Insère un token donné en paramètre à la fin d'une liste simplement chaînée.

insere_jeu() Permet de prélever un token d'une première liste chaînée donnée en paramètre et de l'insérer t au début ou à la fin d'une seconde liste suivant le paramètre direction.

supprime_combo() Fonction récursive qui prend en paramètre un pointeur sur un élément dans une liste et supprime le nombre d'éléments donné en paramètre

repere_combo_couleur/forme() Deux fonctions qui détectent des triplets (ou quadruplets...) de même couleur ou forme d'une liste donnée en paramètre et les supprime grâce à la fonction `supprime_combo()`, elle re-chaîne ensuite les éléments simplement.

chainage_forme_couleur() Chaîne doublement une liste de tokens par forme et par couleur

echange_forme/couleur() Deux fonctions permettant de décaler vers la gauche la forme ou la couleur d'un tokens donné en paramètre.

affiche_liste_g() Permet d'afficher la liste des tokens aléatoire ou celle des tokens placés par le joueur selon l'argument donné.

detecte_clic_forme() Détecte un clic de joueur sur un des tokens placé et lui propose de décaler les tokens de même forme ou même couleur vers la gauche.

2. Difficultés rencontrées

Les difficultés majeures ressenties sur l'ensemble de la conception du programme résidaient dans la manipulation des listes chaînées (chaînage, détection, suppression, désallouage de la mémoire...). Les membres du groupe étant novice dans la langage C, les différences notables entre les langages précédemment étudiés (python...) et le C (comme la manipulation de la mémoire) ont eu tendance à ralentir légèrement l'avancement du projet.

Rapport rédigé par Pierrejean BESNARD et Louis BILLAUT élèves du tda.