



PROJET IAP

Poursuite par équipes en patinage de vitesse

Dans la poursuite par équipes, deux équipes formées chacune de 3 athlètes partent simultanément de deux points opposés de la piste circulaire et doivent faire un nombre t de tours. Le temps réalisé par une équipe est celui obtenu par le dernier patineur de l'équipe. À chaque tour de piste, les temps des deux équipes sont relevés et permettent de suivre l'évolution de la course. Une compétition de poursuite par équipes consiste à organiser n poursuites par équipes permettant de classer $2 \times n$ équipes en ayant comme critère le temps des équipes sur la ligne d'arrivée (après t tours).

Le but du projet est de programmer un interpréteur de commandes permettant (1) de définir le nombre de tours de la poursuite par équipes, (2) de définir le nombre d'équipes, (3) d'inscrire les trois patineurs d'une équipe, (4) d'organiser l'ensemble des épreuves de poursuites par équipes, (5) d'enregistrer pour chaque épreuve par équipes les temps intermédiaires enregistrés à chaque tour, et (6) de calculer les classements intermédiaires entre les deux équipes engagées dans l'épreuve de poursuite et le classement final de la compétition.

L'application doit interpréter huit commandes. Celles-ci sont représentées sous forme de chaînes de caractères non accentués de formats spécifiés dans ce document. Le caractère séparateur entre la commande et les différents champs d'information est le caractère espace. La taille maximale des chaînes de caractères est de 30. Les commandes pourront être soit entrées en utilisant l'entrée standard (le clavier) ou par redirection d'un fichier texte sur l'entrée standard.

Les commandes sont les suivantes :

C1. Commande de sortie du programme

Une ligne composée de la chaîne de caractères "exit".

C2. Commande de définition de l'épreuve de poursuite

Une ligne composée de la chaîne de caractères "definir_parcours" suivie du nombre de tours t (entier supérieur ou égal à 2 et inférieure ou égal à 10).

C3. Commande de définition du nombre d'épreuves de poursuite

Une ligne composée de la chaîne de caractères "definir_nombre_epreuves" suivie du nombre d'épreuves de poursuite n (entier supérieur ou égal à 1 et inférieure ou égal à 16).

C4. Commande d'inscription d'une équipe

Une ligne composée de la chaîne de caractères "inscrire_equipe" suivie du nom du pays et des trois noms des membres de l'équipe. Les numéros de dossard commencent à 101 et seront automatiquement attribués par programme dans l'ordre séquentiel d'inscription. Le nombre maximal d'équipe est de 32. La commande affiche sur la sortie standard "inscription dossard" suivie des trois numéros de dossards attribués.

C5. Commande d'affichage des équipes de l'épreuve de poursuite

Une ligne composée de la chaîne de caractères "afficher_equipes". Cette commande provoquera l'affichage de deux lignes (une par équipe) correspondant aux deux équipes de l'épreuve de poursuite en cours. Chaque ligne sera composée du nom du pays, des trois noms des membres de l'équipe et de leurs numéros respectifs de dossard. L'ordre d'affichage est celui de l'enregistrement des équipes.

C6. Commande d'enregistrement d'un temps chronométré pour un nombre de tours de piste effectué

Une ligne composée de la chaîne de caractères "enregistrer_temps" suivie du numéro de dossard, du nombre de tours de piste effectué et du temps chronométré (un réel). Les numéros de dossard sont ceux des patineurs participant à l'épreuve de poursuite en cours.

Il n'y a pas d'affichage sur la sortie standard. Les enregistrements sont faits dans l'ordre chronologique.

La détection de la fin de la poursuite est faite à chaque enregistrement. Elle intervient si tous les patineurs ont effectué t tours de piste. En cas de détection de fin de poursuite, "detection_fin_poursuite" est affiché, puis la commande d'affichage du temps final des deux équipes. L'ordre d'affichage des équipes est fait dans l'ordre croissant des temps finaux (équipe gagnante de la poursuite en premier).

La détection de la fin de la compétition de poursuite est faite également à chaque enregistrement. Elle intervient si toutes les épreuves de poursuite ont eu lieu. En cas de détection de la fin de la compétition, "detection_fin_competition" est affiché, puis la commande d'affichage du classement de l'ensemble des équipes en fonction du temps final réalisé par chaque équipe, enfin la commande "exit" est appelée.

C7. Commande d'affichage des temps chronométrés d'un patineur

Une ligne composée de la chaîne de caractères "afficher_temps" suivie du numéro du dossard. Cette commande donne les temps chronométrés disponibles dans les enregistrements pour le patineur. Elle affiche sur la sortie standard, une ligne pour chaque temps chronométré dans l'ordre chronologique. Chaque ligne a la forme suivante : nom du pays, numéro du tour effectué, nom du patineur et temps chronométré sous forme d'un réel avec une précision d'un chiffre après la virgule.

C8. Commande d'affichage du temps des équipes pour un nombre de tour donné

Une ligne composée de la chaîne de caractères "afficher_temps_equipes" suivie du nombre de tours donné. Cette commande affiche pour les deux équipes engagées dans l'épreuve de poursuite le nom du pays suivi du temps effectué par le dernier patineur de cette équipe. L'ordre d'affichage est celui de l'enregistrement des équipes. Dans le cas où ce dernier patineur n'a pas encore effectué le nombre de tours donné, « indisponible » est affiché.

L'automatisation des tests pourra se faire par une redirection d'un fichier texte sur l'entrée standard pour les entrées et par une redirection de la sortie standard sur un fichier texte. La fonction `main()` du programme sera composée d'une boucle infinie permettant d'appeler séquentiellement l'une de ces sept fonctions jusqu'à la commande de sortie de programme.

Cadre du développement logiciel

Le projet est à réaliser en binôme (les monômes ne sont pas autorisés). Les membres d'un binôme seront de préférence du même groupe. Vous devez programmer et tester l'application demandée.

Le développement logiciel se fera par cycle de type agile au moyen de *sprints* (5 au total). Chaque sprint est défini par une spécification et un test par redirection correspondant à un jeu de données de test (JDT) (*in.txt*) et ses résultats attendus (*out.txt*). Pour un *sprint* donné, si le résultat de votre application (*outAppliSpn.txt*) coïncide avec le résultat de référence (*outSpn.txt*), votre application est 0-défaut et le *sprint* est validé. Vous passerez alors au développement du *sprint* suivant.

Vous trouverez en Annexe 2 un exemple des fichiers *in.txt* et *out.txt* correspondant au *Sprint#4*.

Votre travail sera évalué à partir (1) d'une recette et (2) de la rédaction du dossier de développement logiciel.

Recette de l'application

La semaine du **21 octobre 2019**, vous passerez la **recette** de votre développement logiciel lors de votre séance **IAP3**. Il s'agit d'un test de recette qui testera le *sprint* de plus haut niveau atteint au cours de votre développement avec un nouveau JDT. Vous aurez à compiler et à exécuter votre programme sur le JDT de recette (type *in.txt*) qui vous sera communiqué. Votre enseignant vérifiera automatiquement le 0-défaut de votre application en comparant le fichier *outAppli.txt* de votre application au fichier de référence (*out.txt*). Si les 2 fichiers coïncident, le sprint est validé à la recette.

Dossier du développement logiciel

Vous devez porter une attention particulière à la rédaction de votre dossier. Sa qualité est déterminante pour l'évaluation de votre travail. La composition de votre dossier doit être la suivante :

- Une page de garde indiquant le **nom** et le **groupe** des membres du **binôme**, l'objet du dossier. Une table des matières paginée sur l'ensemble du dossier incluant les annexes.
- Une brève présentation du projet (1 page) : le rôle fonctionnel de l'application, les entrées et sorties de l'application.
- L'organisation des tests de l'application et le **bilan de validation** des différents *sprints* que vous avez développés.
- Un bilan de projet (les difficultés rencontrées, ce qui est réussi, ce qui peut être amélioré).
- En annexe au dossier :
 - 1) Le listing complet de vos sources.
 - 2) la trace d'exécution du test du *sprint* de plus haut niveau atteint et éventuellement les fichiers *in.txt* et *out.txt* de vos propres tests.

Recommandations

Les **codes sources** doivent être **commentés** (structures de données, champs, fonctions). Les conventions de nommage données en cours doivent être respectées. Les **préconditions** des fonctions doivent être **documentées** et **testées par assertion** dans le code.

Suivez toutes les **spécifications données** sous peine de pénalisation. En particulier, la composition du dossier de développement logiciel et la constitution de l'archive *.zip* demandée.

ATTENTION : Un dossier sans sources sera fortement pénalisé

Date limite de remise de projet

La date limite de remise de projet est fixée au lundi **4 novembre 2019**.

- Déposer la **version papier** (brochée) de votre projet au secrétariat.
- Déposer dans le **puits** (pour chacun des groupes du binôme **IAP/Gr?**) l'archive (**.zip**) de nom « **Nom1Gr?Nom2Gr?.zip** » constituée du dossier de développement (**.pdf**) et du **source** (**sans** l'exécutable **.exe**) du *sprint* de plus haut niveau que vous avez validé lors de la recette.

Annexe 1. Jeu de données de test (*in.txt*) et sortie attendue (*out.txt*)

inSprint1.txt

```
definir_parcours 2
definir_nombre_épreuves 2
inscrire_equipe Canada Blondin Weidemann Morrison
inscrire_equipe Japon Takagi Sato Takagu
afficher_equipes
exit
```

outSprint1.txt

```
inscription dossard 101
inscription dossard 102
inscription dossard 103
inscription dossard 104
inscription dossard 105
inscription dossard 106
Canada Blondin 101 Weidemann 102 Morrison 103 Japon Takagi 104 Sato 105
Takagu 106
```

inSprint2.txt

```
definir_parcours 2
definir_nombre_épreuves 2
inscrire_equipe Canada Blondin Weidemann Morrison
inscrire_equipe Japon Takagi Sato Takagu
enregistrer_temps 101 1 53.1
enregistrer_temps 102 1 53.2
enregistrer_temps 104 1 53.3
enregistrer_temps 105 1 53.7
enregistrer_temps 106 1 53.9
enregistrer_temps 103 1 54.1
enregistrer_temps 105 2 100.6
afficher_temps 102
afficher_temps 105
exit
```

outSprint2.txt

```
inscription dossard 101
inscription dossard 102
inscription dossard 103
inscription dossard 104
inscription dossard 105
inscription dossard 106
Canada 1 Weidemann 53.2
Japon 1 Sato 53.7
Japon 2 Sato 100.6
```

inSprint3.txt

definir_parcours 2
definir_nombre_épreuves 2
inscrire_equipe Canada Blondin Weidemann Morrison
inscrire_equipe Japon Takagi Sato Takagu
enregistrer_temps 101 1 53.1
enregistrer_temps 102 1 53.2
enregistrer_temps 104 1 53.3
enregistrer_temps 105 1 53.7
enregistrer_temps 106 1 53.9
enregistrer_temps 103 1 54.1
enregistrer_temps 105 2 100.6
afficher_temps 102
afficher_temps 105
exit

outSprint3.txt

inscription dossard 101
inscription dossard 102
inscription dossard 103
inscription dossard 104
inscription dossard 105
inscription dossard 106
Canada 54.1
Japon 53.9

inSprint4.txt

definir_parcours 2
definir_nombre_épreuves 2
inscrire_equipe Canada Blondin Weidemann Morrison
inscrire_equipe Japon Takagi Sato Takagu
enregistrer_temps 101 1 53.1
enregistrer_temps 102 1 53.2
enregistrer_temps 104 1 53.3
enregistrer_temps 105 1 53.7
enregistrer_temps 106 1 53.9
enregistrer_temps 103 1 54.1
enregistrer_temps 105 2 100.6
enregistrer_temps 106 2 101.7
enregistrer_temps 104 2 102.3
enregistrer_temps 101 2 102.5
enregistrer_temps 103 2 102.8
enregistrer_temps 103 2 103.1
exit

outSprint4.txt

inscription dossard 101
inscription dossard 102
inscription dossard 103
inscription dossard 104
inscription dossard 105
inscription dossard 106
detection_fin_poursuite
Japon 102.3
Canada 103.1

inSprint5.txt

definir_parcours 2
definir_nombre_épreuves 2
inscrire_equipe Canada Blondin Weidemann Morrison
inscrire_equipe Japon Takagi Sato Takagu
inscrire_equipe France Pierron Huot Monvoisin
inscrire_equipe Italie Lollobrigida Mascitto Valcepina
enregistrer_temps 101 1 53.1
enregistrer_temps 102 1 53.2
enregistrer_temps 104 1 53.3
enregistrer_temps 105 1 53.7
enregistrer_temps 106 1 53.9
enregistrer_temps 103 1 54.1
enregistrer_temps 105 2 100.6
enregistrer_temps 106 2 101.7
enregistrer_temps 104 2 102.3
enregistrer_temps 101 2 102.5
enregistrer_temps 103 2 102.8
enregistrer_temps 103 2 103.1
enregistrer_temps 111 1 50.9
enregistrer_temps 108 1 52.1
enregistrer_temps 112 1 53.2
enregistrer_temps 107 1 53.5
enregistrer_temps 109 1 53.8
enregistrer_temps 110 1 54.1
enregistrer_temps 110 1 99.1
enregistrer_temps 109 1 100.3
enregistrer_temps 107 1 101.5
enregistrer_temps 112 1 101.8
enregistrer_temps 108 1 102.1
enregistrer_temps 111 1 102.6

outSprint5.txt

inscription dossard 101
inscription dossard 102
inscription dossard 103
inscription dossard 104
inscription dossard 105
inscription dossard 106
inscription dossard 107
inscription dossard 108
inscription dossard 109
inscription dossard 110
inscription dossard 111
inscription dossard 112
detection_fin_poursuite
Japon 102.3
Canada 103.1
detection_fin_poursuite
France 101.5
Italie 102.6
detection_fin_competition
France 101.5
Japon 102.3
Italie 102.6
Canada 103.1