

BUT1 – SAÈ S1.01

IMPLÉMENTATION D'UN BESOIN CLIENT

– GESTION DES ABSENCES –

L'objectif du projet est de développer un interpréteur de commande gérant les absences des étudiants d'une formation BUT.

1 Règles de gestion (simplifiées)

Les étudiants inscrits en BUT ne doivent pas cumuler plus de 5 demi-journées d'absences non-justifiées par semestre sous peine de se voir déclarés défaillants (avec pour conséquence que toutes les UE du semestre en cours soient invalidées).

Un étudiant ayant été absent à une demi-journée dispose de 3 jours après l'absence pour déposer un justificatif. Ce justificatif est validé (ou pas) par le directeur d'études.

Une absence pour laquelle aucun justificatif a été déposé dans les temps ou que celui-ci ait été invalidé par le directeur d'études est considérée comme étant non-justifiée (et ne pourra plus l'être).

2 Cahier des charges

L'application que vous devez développer doit interpréter 9 commandes distinctes. Celles-ci sont représentées par des chaînes de caractères de formats spécifiés ci-dessous. Le caractère séparateur entre la commande et les différents champs d'information est le caractère 'espace'. Toute commande se termine par un retour à la ligne.

Votre programme doit lire les commandes sur l'entrée standard et produire des résultats sur la sortie standard (l'écran). Les commandes sont supposées respecter le format attendu. Seules les erreurs explicitement indiquées dans ce document doivent être prises en compte par votre programme.

Les commandes sont les suivantes :

C0 - Sortie du programme

Une ligne composée de la chaîne de caractères `"exit"`. Elle doit provoquer la fin de l'exécution du programme.

C1 - Inscription

Une ligne composée de la chaîne de caractères `"inscription"` suivie du nom de l'étudiant et d'un numéro de groupe. Le nom est une chaîne de caractères ne dépassant pas 30 caractères et ne contenant pas d'espace. Le numéro de groupe est un simple entier.

Votre programme doit afficher le message `"Inscription enregistrée"` suivi d'un identifiant entre parenthèses. Celui-ci doit correspondre au nombre d'étudiants inscrits (1 pour le premier étudiant, 2 pour le suivant, ...).

Le cas échéant, votre programme doit afficher le message `"Nom incorrect"` si un étudiant déjà inscrit dans ce groupe porte le même nom.

Dans les exemples ci-dessous, Le texte en **rouge** correspond aux données saisies par l'utilisateur alors que le texte en **bleu** correspond aux messages affichés par le programme.

```
inscription Lea 101
Inscription enregistrée (1)
```

Le message affiché nous indique que cette étudiante est la première à être inscrite et qu'en conséquence, elle sera désignée par la suite avec l'identifiant 1.

C2 - Enregistrement d'une absence

Une ligne composée de la chaîne de caractères `"absence"` suivie de l'identifiant de l'étudiant, du numéro de jour et de la demi-journée concernée. L'identifiant est un entier devant désigner un étudiant déjà inscrit. Le numéro de jour est un entier compris entre 1 et 40 (un semestre comporte au plus 40 jours). La demi-journée est désignée par la chaîne `"am"` (le matin) ou `"pm"` (l'après-midi).

Votre programme doit afficher le message **"Absence enregistrée"** suivi d'un identifiant entre crochets. Celui-ci doit correspondre au nombre total d'absences connues (1 pour la première, 2 pour la suivante, ...). La nouvelle absence est considérée comme étant en attente de justificatif durant les trois jours qui suivent et comme non-justifiée après.

Le cas échéant, votre programme doit afficher l'un des messages suivants : **"Identifiant incorrect"** s'il ne désigne pas un étudiant, **"Date incorrecte"** si celui-ci n'a pas une valeur autorisée ou, **"Demi-journée incorrecte"** pour les mêmes raisons. Enfin, il doit afficher **"Absence déjà connue"** si c'est le cas.

```
absence 1 1 am
Absence enregistrée [1]
```

C3 - Liste des étudiants

Une ligne composée de la chaîne de caractères **"etudiants"** suivie du numéro de jour courant. Ce dernier est un entier supérieur ou égale à 1 (*i.e.* la date courante peut dépasser la fin du semestre).

Le cas échéant, votre programme doit afficher le message **"Date incorrecte"** ou **"Aucun inscrit"**. Sinon, votre programme doit afficher la liste des étudiants inscrits, en précisant pour chacun son identifiant entre parenthèses, son nom, son groupe et son nombre total d'absences. Pour ce total, seules les absences antérieures (ou égales) au jour courant doivent être prises en compte. Les étudiants doivent être triés par groupe croissant et, au sein d'un même groupe, par ordre alphabétique croissant de leur nom.

```
etudiants 40
(3) Alex 30 4
(2) John 101 10
(1) Lea 101 1
```

Les alignements montrés dans l'exemple ci-dessus peuvent ne pas être respectés par votre programme mais seront considérés comme un plus dans la notation. Les noms sont alignés à gauche, les groupes et les nombres d'absences sont alignés à droite.

C4 - Dépôt d'un justificatif

Une ligne composée de la chaîne de caractères **"justificatif"** suivie de l'identificateur d'une absence, d'un numéro de jour et d'une chaîne de caractères expliquant l'absence. Cette chaîne fait au plus 50 caractères et peut contenir des espaces.

Si le justificatif est déposé dans l'intervalle de temps légal, l'absence est considérée comme étant en cours de traitement et cela tant que le directeur d'études n'a pas étudié le justificatif. Dans le cas contraire, le justificatif est enregistré mais l'absence est définitivement non-justifiée.

Le cas échéant, votre programme doit afficher l'un des messages suivants : **"Identifiant incorrect"** s'il ne désigne pas une absence ou **"Date incorrecte"** si le numéro de jour précède la date de l'absence. Il doit afficher **"Justificatif déjà connu"** si c'est le cas. Sinon, il doit afficher **"Justificatif enregistré"**.

```
justificatif 1 3 j'avais mal au ventre
Justificatif enregistré
```

C5 - Liste des absences en attente de validation

Une ligne composée de la chaîne de caractères **"validations"**.

Le cas échéant, votre programme doit afficher le message **"Aucune validation en attente"**. Dans le cas contraire, il doit afficher la liste des absences pour lesquelles un justificatif a été déposé dans les temps et que celui-ci n'a pas encore été validé ou invalidé par le directeur d'études. Pour chaque absence, votre programme doit afficher :

- l'identifiant de l'absence entre crochets,
- l'identifiant de l'étudiant entre parenthèses, son nom et son groupe,
- La date de l'absence (le numéro du jour et le code de la demi-journée séparés par un **"/"**),
- Le texte du justificatif (entre parenthèses)

Les absences doivent être triées par numéro d'étudiant croissant et, pour un même étudiant, par ordre chronologique.

validations

```
[5] (1) Lea 101 12/am (peu d'interet pour la matiere)
[3] (3) Alex 30 4/pm (je ne savais pas qu'il y avait cours)
[6] (3) Alex 30 6/am (certificat angine)
[8] (3) Alex 30 6/pm (certificat rhume)
[10] (3) Alex 30 8/am (permis de conduire)
```

Les alignements montrés dans l'exemple peuvent ne pas être respectés par votre programme mais seront considérés comme un plus dans la notation. Tous les champs sont alignés sur la gauche à l'exception des numéros de groupe et des dates des absences qui sont alignés sur la droite.

C6 - Validation/invalidation d'un justificatif

Une ligne composée de la chaîne de caractères "validation" suivie de l'identifiant d'une absence et d'une chaîne de caractères indiquant si le justificatif est accepté ("ok") ou pas ("ko").

Votre programme doit afficher le message "Validation enregistrée" et l'absence est considérée comme étant justifiée ou non-justifiée selon le verdict du directeur d'études.

Le cas échéant, votre programme doit afficher le message "Identifiant incorrect" si l'identifiant ne désigne pas une absence connue ou que celle-ci ne soit pas en attente de validation ou encore le message "Code incorrect" si le code de validation est différent de "ok" et "ko", ou enfin le message "Validation déjà connue" si c'est le cas.

```
validation 1 ko
Validation enregistrée
```

C7 - Situation d'un étudiant

Une ligne composée de la chaîne de caractères "etudiant" suivie de l'identifiant d'un étudiant et du numéro de jour courant.

Le cas échéant, votre programme doit afficher l'un des messages suivants : "Identifiant incorrect" si le numéro ne désigne pas un étudiant inscrit ou, "Date incorrecte" si celle-ci est inférieure à 1. Dans le cas contraire, il doit afficher une première ligne décrivant l'étudiant (selon le même format que pour la commande C3) suivie de la liste des absences de l'étudiant classées selon leur statut :

- Les absences pour lesquelles un justificatif peut encore être déposé (le titre de la section est "En attente justificatif")
- Les absences pour lesquelles un justificatif a été déposé et en attente d'une validation par le directeur d'études ("En attente validation")
- Les absences justifiées ("Justifiées")
- Les absences non-justifiées ("Non-justifiées")

Au sein de chaque section, les absences sont affichées une par ligne en indiquant son identifiant (entre crochets), la date de l'absence (le numéro du jour et le code de la demi-journée séparés par un "/") et le texte du justificatif entre parenthèses si celui-ci est connu. Ces absences doivent être triées chronologiquement.

Seules les absences ainsi que les justificatifs (et leur validation éventuelle par le directeur d'études) étant antérieures à la date du jour doivent être pris en compte. Les sections vides ne doivent pas être affichées.

Les alignements sont réalisés section par section. Votre programme peut ne pas les respecter.

```
etudiant 3 20
(3) Alex 30 6
- En attente justificatif
  [5] 1/am
  [6] 3/pm
- En attente validation
  [7] 5/am (j'ai eu mal a la tete)
- Justifiees
  [12] 4/pm (j'ai eu mal au ventre)
- Non-justifiees
  [2] 8/am (j'ai eu mal au ventre)
  [10] 12/am
```

C8 - Liste des étudiants défaillants

Une ligne composée de la chaîne de caractères "defaillants" suivie du numéro de jour courant.

Le cas échéant, votre programme doit afficher le message "Date incorrecte" si celle-ci est inférieure à 1 ou "Aucun defaillant" si tous les étudiants ont moins de 5 absences non-justifiées à ce jour. Sinon, il doit afficher la liste des étudiants atteignant ou dépassant cette limite. Cette liste doit respecter le même format que celui de la commande C3. Notez bien que, dans cette liste, le nombre d'absences correspond au nombre total d'absences et non pas uniquement celles étant non-justifiées.

Ici encore, seules les absences ainsi que les justificatifs étant antérieurs à la date du jour doivent être pris en compte pour déterminer qui est defaillant et qui ne l'est pas.

```
defaillants 50
```

```
(1) Lea 201 15
```

3 Limites numériques

Le nombre d'étudiants est limité à 100. Les noms des étudiants ne dépassent pas 30 caractères, les textes des justificatifs sont limités à 50 caractères. Pour rappel, les dates étiquetant les absences sont comprises entre 1 et 40. Toutes les autres dates employées dans les commandes doivent avoir une valeur supérieure ou égale à 1.

4 Qui, quoi et quand?

Le projet est à réaliser en binôme (les monômes peuvent être autorisés exceptionnellement si vous le demandez). 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 quatre sprints. Chaque sprint est défini par un ensemble de commandes devant être supporté par votre programme et un test par redirection correspondant à un jeu de données de test (JDT) (`in.txt`) et ses résultats attendus (`out.txt`). Les différents JDT seront déposés sur le disque COMMUN sous peu. Pour un sprint donné, si le résultat de votre application coïncide avec le résultat de référence (`out.txt`), le sprint est considéré comme étant validé. Vous passerez alors au développement du sprint suivant. Vous trouverez en annexe (fig. 1) un exemple d'exécution du programme correspondant au dernier sprint¹. Les commandes à prendre en compte par les différents sprints sont les suivantes :

Sprint	Commandes
SP1	<code>inscription</code> , <code>absence</code> , <code>etudiants</code> et <code>exit</code>
SP2	SP1 + <code>justificatif</code> , <code>validations</code> et <code>validation</code>
SP3	SP2 + <code>etudiant</code>
SP4	SP3 + <code>defaillants</code>

Vous avez le droit de programmer d'autres commandes que celles demandées pour vous faciliter la mise au point du programme. Une commande permettant d'afficher l'ensemble des absences peut être judicieuse. Ces commandes supplémentaires ne seront pas testées lors de la recette (voir ci-dessous). Toutefois, il est fortement recommandé de mettre en avant ce travail dans votre rapport de développement (voir ci-dessous) si c'est le cas. Il en sera tenu compte lors de la notation.

Votre travail sera évalué à partir (1) d'une recette, (2) de la rédaction du dossier de développement logiciel et (3) d'une partie du DST de la ressource R1.01.

La semaine du **21 octobre 2024**, vous passerez la recette de votre développement logiciel lors de la seconde séance de TP. 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 avec le JDT de recette (type `in.txt`) qui vous sera communiqué. Votre enseignant vérifiera automatiquement la correction de votre application en comparant l'affichage produit au fichier de référence (`out.txt`). Si les 2 fichiers coïncident, le sprint sera considéré comme étant validé pour la recette.

Vous devez porter une attention particulière à la rédaction de votre dossier. Sa qualité est importante 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.

¹Les fichiers `in.txt` et `out.txt` correspondant respectivement aux commandes saisies et aux messages affichés sont joints au présent document et sont accessibles dans les signets. Toutefois, tous les *viewers PDF* ne supportent pas l'extraction de fichiers attachés. Acroread et Firefox le supportent.

- Une brève présentation du projet (1 page maximum) : le rôle fonctionnel de l'application, les entrées et sorties de l'application.
- Une présentation succincte des moyens que vous aurez mis en œuvre pour vous assurer de la correction de votre programme.
- Un bilan du projet (les difficultés rencontrées, ce qui est réussi, ce qui peut être amélioré).
- En annexe au dossier :
 1. Le code source (commenté et bien indenté) de votre programme.
 2. la trace d'exécution du test du sprint de plus haut niveau atteint.

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 et, en particulier, la composition du dossier de développement logiciel et la constitution de l'archive .zip demandée ci-dessous sous peine de pénalisation.

La date limite de remise de projet est fixée au **vendredi 25 octobre 2024**. Vous devrez déposer sur MOODLE une archive (.zip) constituée du dossier de développement (.pdf) et du source (uniquement le fichier C) du sprint de plus haut niveau que vous avez validé lors de la recette.

Annexe

<pre> inscription Lea 101 Inscription enregistree (1) inscription John 101 Inscription enregistree (2) inscription Alex 30 Inscription enregistree (3) absence 1 1 am Absence enregistree [1] absence 1 5 pm Absence enregistree [2] absence 3 3 am Absence enregistree [3] absence 3 2 pm Absence enregistree [4] etudiants 80 (3) Alex 30 2 (2) John 101 0 (1) Lea 101 2 etudiants 3 (3) Alex 30 2 (2) John 101 0 (1) Lea 101 1 justificatif 1 5 j'avais mal au ventre Justificatif enregistre justificatif 2 5 j'avais piscine Justificatif enregistre justificatif 4 5 j'avais une garde a vue Justificatif enregistre </pre>	<pre> validations [2] (1) Lea 101 5/pm (j'avais piscine) [4] (3) Alex 30 2/pm (j'avais une garde a vue) validation 2 ko Validation enregistree validation 4 ok Validation enregistree etudiant 1 20 (1) Lea 101 2 - Non-justifiees [1] 1/am (j'avais mal au ventre) [2] 5/pm (j'avais piscine) etudiant 3 5 (3) Alex 30 2 - En attente justificatif [3] 3/am - Justifiees [4] 2/pm (j'avais une garde a vue) etudiant 3 20 (3) Alex 30 2 - Justifiees [4] 2/pm (j'avais une garde a vue) - Non-justifiees [3] 3/am defaillants 20 Aucun defaillant exit </pre>
---	--

Figure 1: Exemple de session – en rouge, les données saisies par l'utilisateur, en bleu, les messages affichés par le programme.