Institut des sciences du Digital
Management & Cognition

UNIVERSITÉ DE LORRAINE

L2 MIASHS – Sciences Cognitives

Projet final

Programmation JAVA

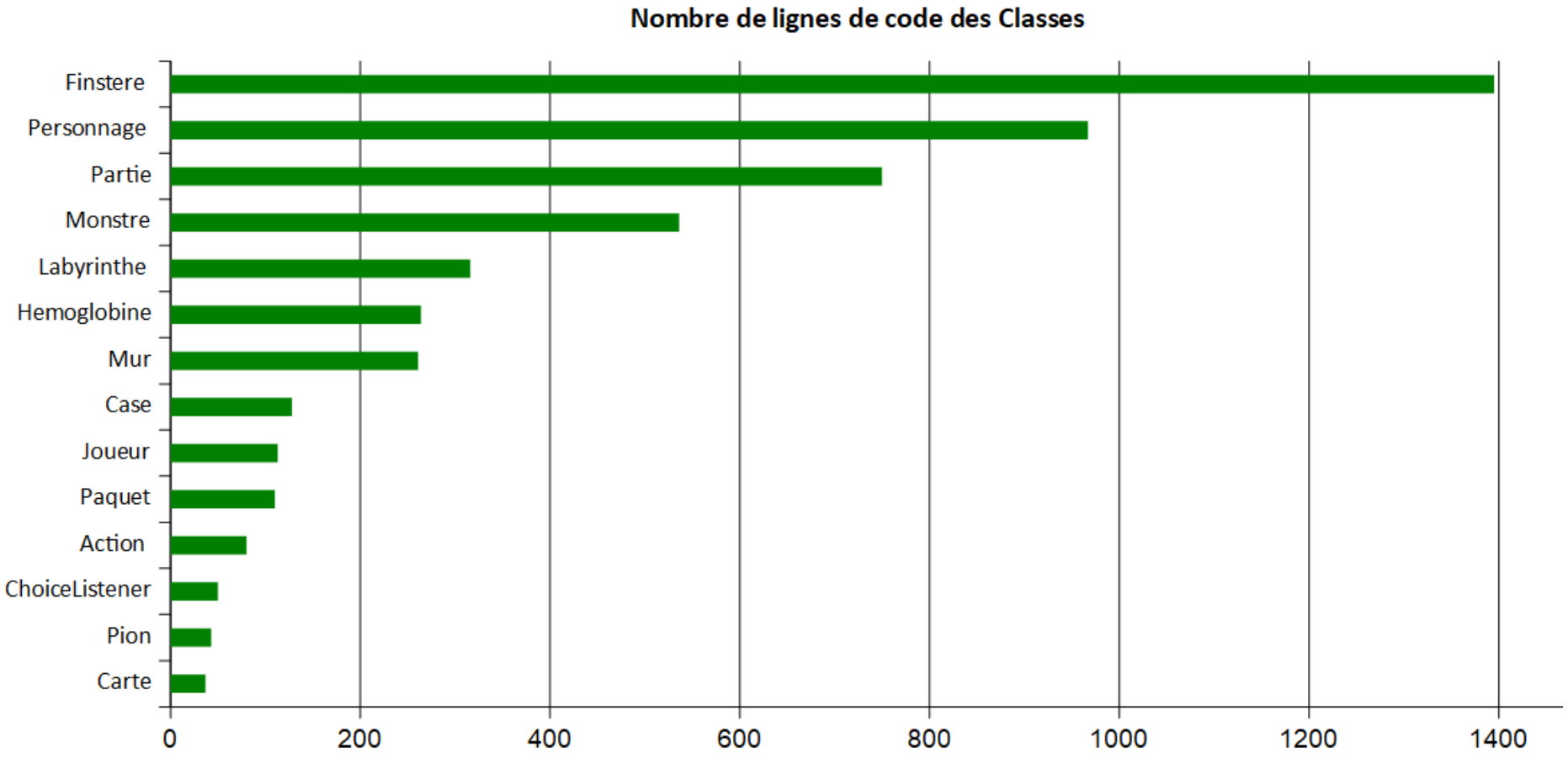
FINSTERE FLURE

CARBONNIER Nicolas – WYSOCKI Tom – WATELET Manon

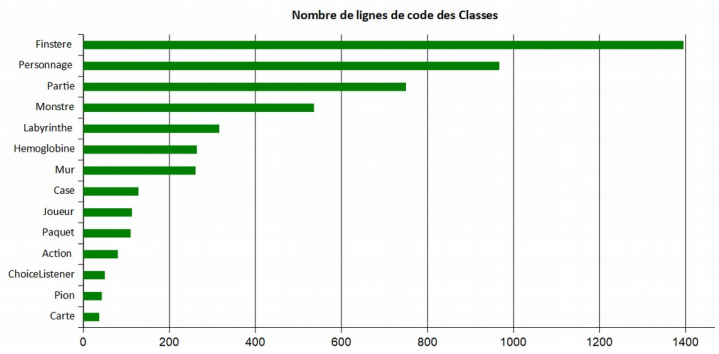
Sommaire

- Programmation
- Le Jeu
  - Terminal
  - Interface
- I.A.
  - Les types
  - Les sous-algos utiles
  - Les algos de stratégie
- Rétrospective
- Démonstration

Le projet JAVA – Programmation



Le projet JAVA – Programmation



Le projet JAVA – Le Jeu : *Terminal*

Output - Finstere_Flure (run)

```
...Génération du plateau de Jeu...

La partie commence !!!
Bonne chance à vous !!!

== MANCHE 1 - TOUR 1 ==

|M>>|   |   |   |   |   |   |   |   |   |   |   |###|###|###|###|
|   |   |   |   |   |   |   |   |   |   |   |   |   |###|###|###|
|   |   | W |   |   |   |   |   | H | H |   |   |   |   |###|###|
|   |   |   |   |   |   |   |   | H | H |   |   | W |   |   |###|
|   |   |   |   |   |   |   | W |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   | W |   |   |   |   | W |   |   |
|   |   |   |   |   |   |   | W |   |   |   |   |   |   |   |   |
|###|   |   |   | W |   |   |   |   |   |   |   | W |   |   |   |
|###|###|   |   | H | H | H | H |   |   |   |   |   |   | W |   |
|###|###|###|   |   | W |   |   | W |   |   |   |   |   |   |   |
|###|###|###|###|   |   |   |   |   |   |   |   |   |   |   |   |
```

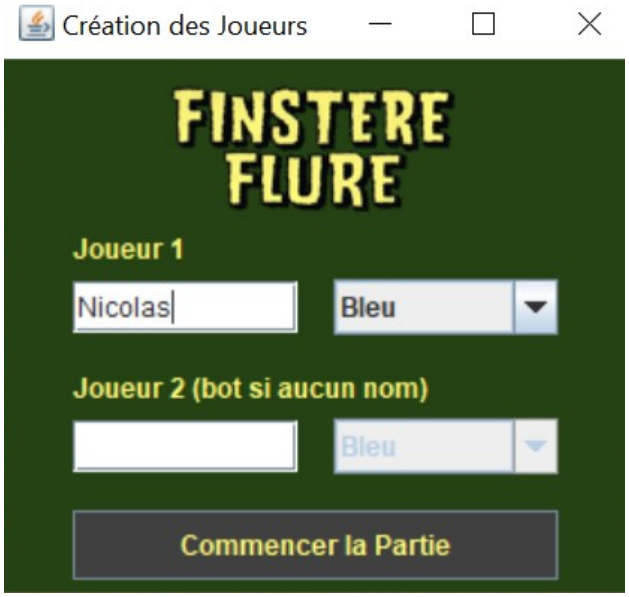
Le projet JAVA – Le Jeu : *Terminal*

```
Pierre :
   Personnage Bleu 1 Face claire (Extérieur)
   Personnage Bleu 3 Face claire (Extérieur)
   Personnage Bleu 4 Face claire (Extérieur)
   Personnage Bleu 5 Face claire (Extérieur)

Choix du Personnage
   1 => Personnage Bleu 1 Face claire (Extérieur)
   2 => Personnage Bleu 3 Face claire (Extérieur)
   3 => Personnage Bleu 4 Face claire (Extérieur)
   4 => Personnage Bleu 5 Face claire (Extérieur)
Choix : 2

Actions possibles (3 pm restants)
   1 => Se Déplacer en (15,8)
   2 => Se Déplacer en (14,9)
   3 => Se Déplacer en (15,9)
   4 => Se Déplacer en (13,10)
   5 => Se Déplacer en (14,10)
   6 => Se Déplacer en (15,10)
Choix : 3
```

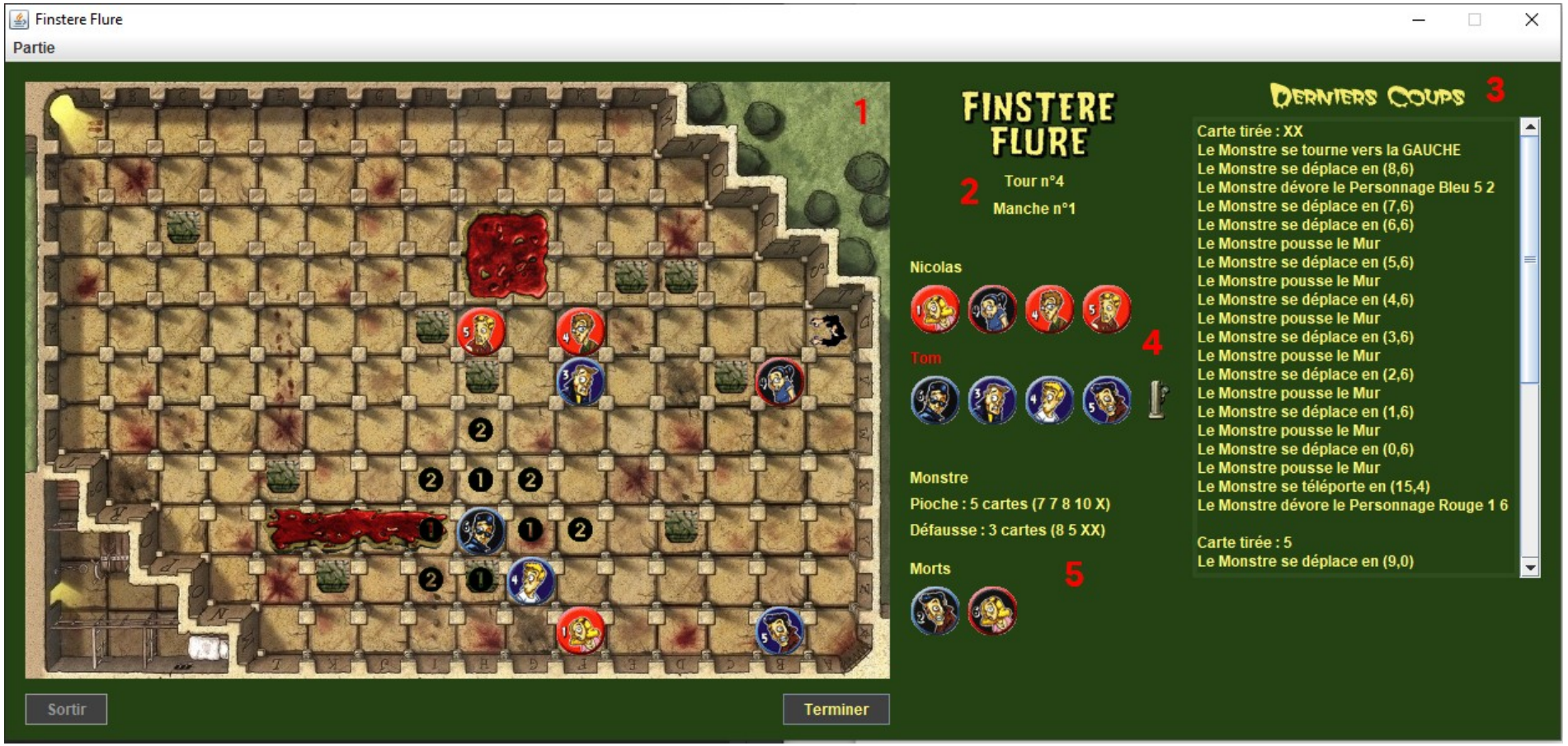
```
...Génération du plateau de Jeu...

La partie commence !!!
Bonne chance à vous !!!

== MANCHE 1 - TOUR 1 ==

|M>>|   |   |   |   |   |   |   |   |   |   |   |###|###|###|###|
|   |   |   |   |   |   |   |   |   |   |   |   |   |###|###|###|
|   |   | W |   |   |   |   |   | H | H |   |   |   |   |###|###|
|   |   |   |   |   |   |   |   | H | H |   |   | W |   |   |###|
|   |   |   |   |   |   |   | W |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   | W |   |   |   |   | W |   |   |
|   |   |   |   |   |   | W |   |   |   |   |   |   |   |   |   |
|###|   |   |   | W |   |   |   |   |   |   |   | W |   |   |   |
|###|###|   |   | H | H | H | H |   |   |   |   |   |   | W |   |
|###|###|###|   |   | W |   |   | W |   |   |   |   |   |   |   |
|###|###|###|###|   |   |   |   |   |   |   |   |   |   |   |   |
```

```
|M>>|   |   |   |   |   |   |   |   |   |   |   |###|###|###|###|
|   |   |   |   |   |   |   |   |   |   |   |   |   |###|###|###|
|   |   | W |   |   |   |   |   | H | H |   |   |   |   |###|###|
|   |   |   |   |   |   |   |   | H | H |   |   | W |   |   |###|
|   |   |   |   |   |   |   | W |   |   |   |   |   |   |   |   |
|   |   |   |   |   |   |   |   | W |   |   |   |   | W |   |   |
|   |   |   |   |   |   | W |   |   |   |   |   |   |   |   |   |
|###|   |   |   | W |   |   |   |   |   |   |   | W |   |   |   |
|###|###|   |   | H | H | H | H |   |   |   |   |   |   | W |   |
|###|###|###|   |   | W |   |   | W |   |   |   |   |   |   |B3C|
|###|###|###|###|   |   |   |   |   |   |   |   |   |   |   |   |
```

Le projet JAVA – Le Jeu : *Interface*



Le projet JAVA – Le Jeu : *Interface*



1 – Plateau de jeu
2 – Informations sur la partie
3 – Derniers coups
4 – Liste des personnages
5 – Infos sur le monstre + liste des morts

Le projet JAVA – I.A.

BUT : Mise en place d'une stratégie pour gagner la partie avec représentation des données

Comment ? Parcours de graphe avec analyse de gain attendu

Moyens :

→ Création des types nécessaires

→ Définition de ce qu'est un état et quelles sont les actions (cf code Java)

→ Création des variables

→ Description des algos et rédaction des SSA **`But?`** Et **`Recherche`**

Le projet JAVA – I.A. : *Les types*

```
Tetat = ARTICLE
          sorti : BOOLEEN  // vrai si pion sauvé
          mort : BOOLEEN  // vrai si pion mort
          dm : NUMERIQUE  // distance par rapport au monstre
          ds : NUMERIQUE  // distance par rapport à la sortie
        FART

Tmap = ARTICLE
          labyrinthe : TABLEAU DE 16 TABLEAU DE 11  // tableau à double entrée
        FART

Tpion = ARTICLE
           e : Tetat
           coord : TABLEAU DE 2 NUMERIQUE  // coordonnées x et y dans le labyrinthe
         FART

Tmonstre = ARTICLE
          coord : TABLEAU DE 2 NUMERIQUE  // coordonnées x et y
          orient : NUMERIQUE
        FART

Tcoup = ARTICLE
          e : Tetat
          p : Tpion
        FART
```

Opérations effectuables pour un pion:

- Tetat SSALGO Déplacer (p: Tpion)
- Tetat SSALGO PousserMur
- Tetat SSALGO Glissade
- BOOLEEN SSALGO CaseOccupée *(par le monstre, un pion, un mur)*

Mise en place de la recherche:

- Tliste SSALGO PionDispo *(liste des pions disponibles au moment de jouer un coup)*
- NUMERIQUE SSALGO Distance

Stratégie de recherche:

- BOOLEEN SSALGO But? *(retourne VRAI si une solution intéressante est trouvée)*
- Tliste SSALGO Successeur
- Tliste SSALGO Recherche

```
BOOLEEN     SSALGO      But? (p : Tpion PAR REFERENCE)

Variable

      lmort : Tliste de Tpion
      ds1 : NUMERIQUE //distance par rapport à la sortie au coup n
      ds2 : NUMERIQUE //distance par rapport à la sortie au coup n+1
      dm1 : NUMERIQUE //distance par rapport au monstre au coup n
      dm2 : NUMERIQUE //distance par rapport au monstre au coup n+1

Début

 RETOURNE( Appartient(lmort, p) OU (p.ds1-p.ds2)>0 OU (p.dm1-p.dm2)<0 )

Fin
```

```
l_successeur    SSALGO    Successeur (p : T_pion , cp : T_coup)

Variable

    l_sortie : T_liste de T_pion
    l_mort : T_liste de T_pion
    l_successeur : T_liste de T_coup
    l_défaite : T_liste de T_coup
    ds1 : NUMERIQUE //distance par rapport à la sortie au coup n
    ds2 : NUMERIQUE //distance par rapport à la sortie au coup n+1
    dm1 : NUMERIQUE //distance par rapport au monstre au coup n
    dm2 : NUMERIQUE //distance par rapport au monstre au coup n+1

Début
/
Fin
```

```
lvictoire      SSALGO      Recherche (lsuccesseur PAR REFERENCE)

Variable

      lvictoire : Tliste de Tcoup
      lneutre : Tliste de Tcoup
      coup : Tcoup

Début

  lvictoire <- CREER()
  lneutre <- CREER()
  TANT QUE (NON VIDE(lsuccesseur)) FAIRE
    | coup <- TETE(lsuccesseur)
    | lsuccesseur <- RESTE(lsuccesseur)
    | SI (But?(coup)) ALORS
    |    |    lvictoire <- ADJ (lvictoire , coup)
    |    |    SINON
    |    |    lneutre <- ADJ (lneutre , coup)
    | FSI
  FTQUE
  SI (NON VIDE (lvictoire) ALORS
    | RETOURNE (lvictoire)
    | SINON
    | RETOURNE (lneutre)
  FSI

Fin
```

Rétrospective

Jeu totalement fonctionnel
Défis 1, 2 et 3 effectués
Bonne communication du groupe

Points forts de l'application:

- Interface ergonomique
- Possibilité de jouer dans le terminal ou dans l'interface

Points faibles :

- Certains bugs mineurs
- Absence de l'IA dans le jeu

Démonstration

Merci pour votre attention