

Projets Dev DII3 2017-18

#1 calcul d'itinéraire dans un graphe, approche multicritere

Encadrant : [Emmanuel Néron\(emmanuel.neron@univ-tours.fr\)](mailto:emmanuel.neron@univ-tours.fr)

Descriptif : Ce projet s'inscrit dans la continuité d'un travail qui a permis de développer un GPS pour vélos (cf <http://www.geovelo.fr/>). Il s'agit ici de développer des algorithmes permettant de calculer le meilleur itinéraire tout en optimisant un ensemble de critères (fonctions objectifs). A noter que chercher le meilleur chemin optimisant un seul critère est facile à résoudre (des algorithmes efficaces existent). Cependant, ce n'est plus le cas dès lors plusieurs critères sont à considérer.

Objectif :

proposer un algorithme de calcul d'itinéraire efficace, l'implémenter puis effectuer une analyse fine des performances de votre algorithme (qualité de la solution/temps de calcul).

Langage : Python (de préférence) ou C

=====

#2 Parsing XLS/CSV et consolidation de services

Encadrant : [Carl Esswein \(carl.esswein@univ-tours.fr\)](mailto:carl.esswein@univ-tours.fr)

Descriptif : La direction des études et la scolarité maintiennent à jour un fichier XLSX avec, pour chaque enseignement, les affectations prévisionnelles des enseignants et des intervenants extérieurs. Parfois une seule personne prend en charge tout l'enseignement, parfois l'enseignement est réparti entre deux ou trois personnes. Parfois une personne prend les CM et une autre les TD, parfois seuls les TD sont répartis – parfois horizontalement parfois verticalement... Ce fichier a beaucoup d'utilités mais il a pour défaut criard de ne pas fournir de vue par personne.

L'objectif de ce projet est de concevoir et réaliser un programme permettant principalement :

- (1) de parser ce fichier d'affectation et d'afficher les "services" consolidés,
- (2) de permettre un export des services consolidés, par personne.

=====

#3 Suivi des retours de notes de CT

Encadrant : [Carl Esswein \(carl.esswein@univ-tours.fr\)](mailto:carl.esswein@univ-tours.fr)

Descriptif : La direction des études et la scolarité effectuent un suivi informel des retours des remontées de notes par les enseignants. Le pointage des notes "manquantes" se fait à la main et est donc, par définition, source d'omissions ou d'erreurs. L'objectif de ce projet est de concevoir et réaliser un programme permettant de :

- visualiser sous une forme adaptée (vue calendrier?) les dates des différentes épreuves de CT, matière par matière, UE par UE,
- saisir les dates de retour des notes par les enseignants,
- visualiser les notes en attente, par matière, par UE, par enseignant

- effectuer des rappels périodiques des notes en attente (à la scolarité et à la direction des études éventuellement étendue aux resp. d'année),

La liste des CT à prévoir pourra être obtenue en début d'année par parsing de la maquette au format XLSX. Les dates des CT pourront être obtenues directement en interrogeant l'EDT en ligne.

Pas de contrainte de langage. Java serait bienvenue mais ce n'est pas du tout une contrainte forte.

=====

#4 Prise en main de l'eye tracker Tobii 4C

Encadrant : Donatello CONTE (donatello.conte@univ-tours.fr)

Descriptif : Le projet a pour objectif d'implémenter un programme pour la récupération des mouvement oculaires, à travers le dispositif Tobii 4C qui a un SDK fournit par le producteur. Le logiciel à implementer pourrait être un petit jeu contrôlé avec les yeux. Le langage de programmation sera le C++.

=====

#5 Implémentation d'un algorithme pour le calcul de la carte de saillance d'une image ou d'une vidéo

Encadrant : Donatello CONTE (donatello.conte@univ-tours.fr)

Descriptif : Une carte de saillance sur une image est une carte qui montre les zones qui retiennent plus l'attention d'un observateur. Il existe plusieurs algorithmes pour calculer cette carte, étant donnée une image ou une vidéo. L'objectif de ce projet est d'implémenter un de ces algorithmes.

=====

#6 Plateforme de relevés de mesures

Encadrant : Pierre Gaucher (pierre.gaucher@univ-tours.fr)

Descriptif : Réaliser le relevé de la température, humidité et pression atmosphérique et gérer un circuit d'horloge temps réel.

A partir des mesures effectuées (dont la périodicité devra être paramétrée) :

- Stocker ces mesures, avec horodatage, sur une carte SD. Le format de stockage des données devra être précisé.
- Afficher les données sur un écran TFT couleur, avec gestion du rafraichissement.
- Calcul de la variation moyenne des grandeurs effectuées par tranche d'1 heure.

Prévoir un menu pour la mise à jour du circuit RTC depuis l'écran TFT tactile.

A partir des relevés de mesures stockées sur la carte SD, prévoir une application permettant d'afficher des graphiques horodatés de la variation de la température, pression et humidité. Plusieurs modalités peuvent être envisagées :

- Ecrire une appli Windows ;
- Afficher les graphiques sur un terminal Android, en gérant une liaison Wifi ou Bluetooth.

Base matérielle : Arduino + shields Arduino spécifiques

=====

#7 Programmation microcontrôleur en Python

Encadrant : [Pierre Gaucher \(pierre.gaucher@univ-tours.fr\)](mailto:pierre.gaucher@univ-tours.fr)

Descriptif : Sur la base de la carte PyBoard V1.1, réaliser un tutorial qui permet à tout utilisateur :

- D'installer l'IDE de développement ;
- De gérer les ressources matérielles de la carte microcontrôleur ;
- De gérer des ressources périphériques (capteurs, actionneurs, Leds, afficheurs, module GPS, ...).

A partir de la maîtrise de la gestion de différentes ressources, réaliser une application, à définir en cours de projet, qui fera la synthèse des éléments définis ci-dessus.

Base matérielle et logicielle : carte MicroPython V1.1 et IDE Spyder Anaconda Python

Référence : <https://micropython.org/> et

<http://docs.micropython.org/en/latest/pyboard/>

***Remarque :** ce projet peut être prolonger sur un projet de développement électronique, qui consisterait à capitaliser les compétences logicielles pour concevoir et réaliser une base robotique mobile autonome.*

#8 Initiation à la reconnaissance des formes avec des méthodes d'appariement de graphes

Encadrant : [Romain Raveaux \(romain.raveaux@univ-tours.fr\)](mailto:romain.raveaux@univ-tours.fr)

Descriptif :

Le mot *forme* est au sens très général, il ne s'agit pas que de forme géométrique. Les *formes* ou *motifs* à reconnaître peuvent être de natures très variées. Il peut s'agir de contenu visuel (code barre, visage, empreinte digitale...) ou sonore (reconnaissance de parole), d'images médicales (rayon X, EEG, IRM...) ou multispectrales (images satellitaires) et bien d'autres. »

Wikipedia

Les étapes :

- 1°) Prendre en main LocalSolver sur un exemple de test.
<https://www.localsolver.com/documentation/exampletour/toy.html>
- 2°) Prendre en main un outil logiciel d'appariement de graphes développé en C#.
- 3°) Intégrer la résolution par LocalSolver du problème d'appariement de graphe.

Encadrant : Romain Raveaux (romain.raveaux@univ-tours.fr)

4

« La **reconnaissance de formes** (ou parfois **reconnaissance de motifs**) est un ensemble de techniques et méthodes visant à identifier des *motifs* informatiques à partir de *données brutes* afin de prendre une décision dépendant de la catégorie attribuée à ce motif¹. On considère que c'est une branche de l'**intelligence artificielle** qui fait largement appel aux techniques d'**apprentissage automatique** et aux **statistiques**.

Le mot *forme* est au sens très général, il ne s'agit pas que de forme géométrique.

Les *formes* ou *motifs* à reconnaître peuvent être de natures très variées. Il peut s'agir de contenu visuel (code barre, visage, empreinte digitale...) ou sonore (reconnaissance de parole), d'images médicales (rayon X, EEG, IRM...) ou multispectrales (images satellitaires) et bien d'autres. »

Wikipedia

Objectif : Initier un binôme d'étudiants à la **reconnaissance de formes** en leur faisant développer de A à Z un système de classification à base de réseaux de neurones artificiels.

1°) Qu'est ce que la classification

2°) Qu'est ce qu'un neurone artificiel

3°) Comprendre et implémenter un Perceptron

<https://fr.wikipedia.org/w/index.php?title=Perceptron&oldid=121736364>

4°) Visualiser les résultats : données+plan séparateur.

5°) Comprendre et implémenter un Perceptron multi-couches

6°) Vers le deep learning¹ : Implémenter le réseau LeNet5 (voir lien de Yann Lecun).

<http://yann.lecun.com/exdb/lenet/>

Compétences : C/C++, Algèbre Linéaire, Curiosité

=====

#10 Word Mole

Encadrant : Ronan Bocquillon (ronan.bocquillon@univ-tours.fr)

Descriptif : Le *Word Mole* est un jeu de lettres populaire. En début de partie, un plateau de NxN lettres est généré aléatoirement. Le joueur doit alors sélectionner le plus rapidement possible des lettres dans l'ordre de son choix de manière à former un mot. Plus le mot est long, plus le joueur marque de points. Ce mot doit être dans le dictionnaire. Le cas échéant, les lettres utilisées sont remplacées par d'autres lettres ou par une pierre (la case devient alors inaccessible jusqu'à la fin de la partie). En validant un mot, le joueur gagne également du temps de jeu. Le but est de marquer le plus de points possible avant l'expiration du compte à rebours.

L'objectif de ce projet est d'implémenter un *Word Mole* en C à l'aide de la bibliothèque SDL 2.0.

=====

#11 Benchmark pour la détection de duplicata WebTV

Encadrant : Mathieu DELALANDRE (mathieu.delalandre@univ-tours.fr)

Descriptif :

L'image est considérée comme le plus fort vecteur d'influence de la société contemporaine. Ces deux dernières décennies, les canaux de diffusion image ont fortement évolués via le numérique. En dehors des canaux dits réguliers (e.g. câble, ADSL, satellite, etc.), les flux vidéo Internet ont pris une part de marché de plus en plus importante. Les études reportent que les flux vidéo représenteront 80% du trafic internet d'ici 2019. A titre d'illustration, cela représente une durée de 5 millions d'années / mois et un volume de données de 1.6 zettaoctets / an. Ces flux se composent de vidéos hors lignes (e.g. YouTube) mais aussi de flux vidéo dits "live" comme les WebTV. Les flux de WebTV ont fortement augmentés ces dernières années. Ces flux sont diffusés 24h/24h rendant les programmes disponibles à l'instant de la diffusion. Ce sujet s'intéresse à la détection de duplicata de vidéo au sein des flux WebTV. La détection de duplicata vise à détecter des copies de vidéo ou parties de vidéo. Elle présente de multiples applications (protection de droit d'auteur, structuration de flux TV, indexation de vidéo, analyse média, etc.). En particulier, dans le contexte WebTV la détection doit se faire sous contrainte temps-réel liée au mode de diffusion. Dans ce contexte, les méthodes de détection opèrent souvent au niveau "frame". L'objet de ce projet est de mettre en place une base de données benchmark pour la détection de duplicata de vidéo WebTV. Différents aspects pourront être abordés (1.), (2.) et/ou (3.)

1. mise en place d'un script de segmentation vidéo (e.g. outil ffmpeg),
 2. constitution d'une base de données à partir des archives publiques <https://archive.org/> , ou vidéo en ligne libre de droit <http://stream.online-downloader.com/> , mise en place d'une approche de sélection de frame clé [1] et de dégradation,
 3. formatage de base existante [2] pour l'identification des séquences et alignement [1]
- A Survey on Visual Content-Based Video Indexing and Retrieval, TSMC 2011 [2]
<http://www.yugangjiang.info/research/VCDB/> Compétences requises: codage script / shell, langage C

=====

#12 Outil d'aide à la décision pour l'affectation de projets

Encadrant : Ameer SOUKHAL (ameur.soukhal@univ-tours.fr)

Etudiants : LOCHE Jérémy – THOMAS Louis

Descriptif : Cet outil permettra aux apprentis d'établir une liste de vœux de projets à partir d'une liste de sujets proposés inscrit dans une base de donnée. La saisie des vœux pourra se faire via une interface web et/ou une application mobile. L'outil comprends un algorithme de recherche opérationnelle (à développer avec l'outil de saisie) permettant à chacun d'être affecté à un projet le mieux classé dans sa liste de vœux tout en faisant des compromis s'il y a des conflits. Il s'agira d'une aide à la décision permettant de résoudre le problème d'affectation de projets. Facultatif -> (L'outil devra être pensé pour être utilisé par les autres départements de l'école afin qu'il soit un outil universel à l'affectation des projets.)

=====

#13 Objets connectés pour l'habitat

Encadrant : Ameer SOUKHAL (ameur.soukhal@univ-tours.fr)

Etudiants : Hugo POHIER, ???

Descriptif : Les objets connectés dans l'habitat, nous ont permis avec le temps, de pouvoir centraliser toutes les informations et les commandes dans une application complète, sur mobile ou sur ordinateur. Ce qui a permis aux personnes d'éviter des déplacements futiles, en pouvant tout consulter, et agir, depuis un support que l'on a en général tout le temps avec soi. Le but de ce projet est de rassembler des informations issues de capteurs, de les stocker dans une base de donnée accessible sur internet, et de rediriger ces informations vers une application mobile.

Objectifs :

- Mettre en place la récolte des informations de capteurs
- Créer une base de données dans lequel les informations seront stockées
- Rediriger les informations vers cette base de données
- Développer une application mobile
- Connecter l'application à la base de données
- Afficher ces données visuellement dans l'application

Prérequis :

- Concevoir et créer une base de données ☑ Savoir faire des requêtes SQL
- Solides bases dans des langages de programmation élémentaires (C/C++/Java)



#14 DECOUVERTE, MISE EN ŒUVRE ET ALTERNATIVES A XDOTOOL

Encadrant : Alexis Rolland (alexis.rolland@univ-tours.fr)

Descriptif : simuler, à l'aide d'une bibliothèque existante (libxdotool) des événements clavier/souris à partir de trames RS232

Mise en situation, contexte et besoins

Il arrive parfois que l'interaction avec un logiciel standard (traitement de texte, navigateur internet...) installé sur un ordinateur standard via le clavier et/ou la souris soit impossible. Par exemple, dans le cas d'un utilisateur présentant un handicap

physique, ou bien encore dans le cas où l'on ne souhaite pas laisser à la disposition des utilisateurs l'accès à ces périphériques, ou s'ils sont remplacés par des dispositifs d'IHM spécifiques. Une borne d'arcade « maison », ou une table basse connectée peuvent être pris pour exemples :



Le problème rencontré alors est de simuler l'activité clavier/souris afin de piloter le logiciel à partir de l'équipement mis à disposition de l'utilisateur :

- ✓ Ensemble de boutons + joystick
- ✓ Interface vocale
- ✓ Interface gestuelle
- ✓ ...

Les logiciels à piloter étant standard, il est impossible d'adapter leur code à ces interfaces. La solution apparaissant « immédiatement » est le développement d'un pilote (driver) pour le dispositif.

Cette solution souffre d'un certain nombre d'inconvénients :

- ✓ Complexité du travail de développement
- ✓ Souplesse vis-à-vis de divers matériels / OS
- ✓ Ajout de nouvelles fonctionnalités
- ✓ ...

Il s'agirait donc ici d'évaluer une solution, basée sur un outil existant, permettant de simuler les actions clavier/souris à partir d'un dispositif « quelconque »

L'outil Xdotool et la bibliothèque libxdotool

Xdotool est un programme, disponible pour les OS GNU/Linux, permettant de simuler des événements clavier et souris. Il se présente sous forme d'un exécutable auquel les paramètres passés correspondent aux événements à générer. Par exemple :

- ✓ `xdotool key "Return"`
 - Simule l'appui sur la touche « Retour chariot »
- ✓ `xdotool mousemove 100 200`
 - Simule un déplacement souris
- ✓ ...

Utiliser cet outil de cette manière permet notamment de rassembler un ensemble d'actions sous forme de scripts. Présenté comme ça, cet outil ne semble pas spécialement en mesure de répondre aux besoins.

Un aspect plus intéressant de ce logiciel est qu'il utilise les services d'une bibliothèque « libxdotool » qui met à disposition des fonctions permettant de simuler les événements.

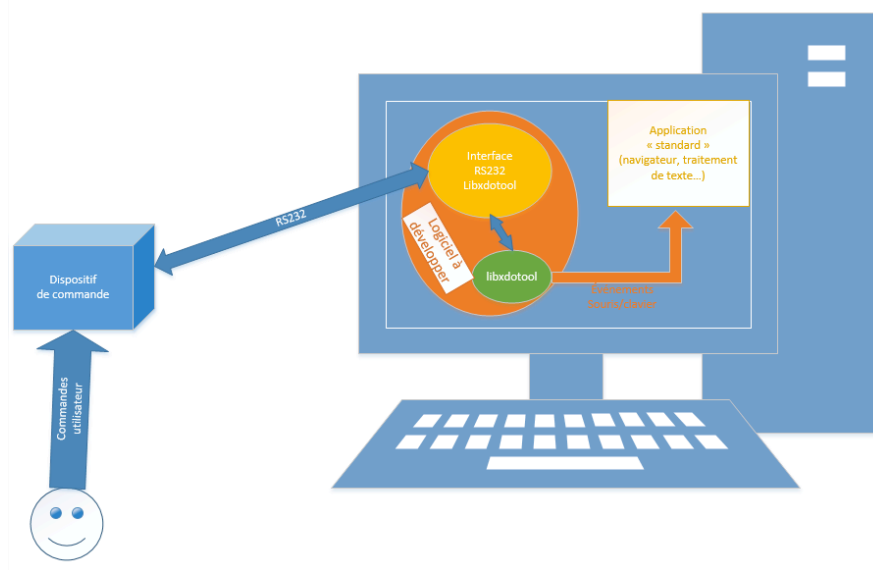
Il est tout à fait possible d'utiliser cette bibliothèque depuis une application développée par nos soins.

L'idée

L'idée est donc de développer un logiciel en C qui interface un dispositif « quelconque » connecté à l'ordinateur avec le système d'exploitation et/ou une application « standard » en se servant de la bibliothèque libxdotool.

Afin de garantir une souplesse de fonctionnement, il faut définir le moyen de communication entre le dispositif et l'ordinateur. Dans un premier temps ce moyen de communication est la liaison série RS232, disponible sur tous les ordinateurs (soit de manière native, soit via un adaptateur USB) et dont la mise en œuvre depuis un logiciel ne pose aucun problème (en termes de faisabilité).

Le synoptique suivant permet de montrer l'architecture du système :



Objectif du projet – Travail à réaliser

L'objectif principal du projet consiste donc à développer une application :

- ✓ Recevant et interprétant des trames RS232, issues du dispositif de commande
- ✓ Traduire les informations protégées par ces trames en événements clavier/souris
- ✓ Générer ces événements à l'aide de la bibliothèque libxdotool

Afin d'atteindre cet objectif, plusieurs tâches seront à réaliser :

- ✓ Installer et prendre en main le logiciel xdotool
- ✓ Mettre en place une communication RS232 basée sur des opérations de lecture/écriture sur un port série standard
- ✓ Etudier et documenter le moyen d'utiliser les fonctions de la bibliothèque libxdotool depuis un programme en C
 - Installation
 - Linkage
 - ...
- ✓ Etudier les fonctions mises à disposition par la bibliothèque, lister et documenter celles qui paraissent le plus évidemment utiles.

- Définir le moyen le plus adéquat (s'il y a le choix) d'utiliser ces fonctions (linkage statique, chargement dynamique)
- ✓ Mettre en place un premier programme simple mettant en évidence le fonctionnement « de bout en bout »
- ✓ Lister les principales fonctionnalités du logiciel à développer
- ✓ Ebaucher un protocole applicatif pour les trames RS232, à partir des principales fonctionnalités définies
- ✓ Analyser et coder le logiciel de manière itérative
 - Ajout des fonctionnalités les unes après les autres
- ✓ Evaluer le système :
 - Utilisabilité
 - Adaptabilité
 - Portabilité
 - ...

Un second objectif du projet, non prioritaire, mais néanmoins à traiter est lié au fait que cette bibliothèque semble n'être disponible que sur des cibles GNU/Linux. Il s'agit donc de rechercher d'autres outils similaires dans la philosophie qui permettraient de porter le système sur d'autres OS (Windows, Mac OS...) et d'en faire une comparaison « a priori ».

Critères de réussite

Le projet sera considéré réussi si les objectifs minimaux suivants sont atteints :

- ✓ Une communication RS232 est possible
- ✓ La bibliothèque libxdotool est utilisable depuis un programme développé en C
- ✓ L'intégration de cette bibliothèque à un projet C est clairement documentée
- ✓ La génération d'un événement clavier/souris suite à la réception d'une trame RS232 est fonctionnelle
- ✓ Une liste d'alternatives à cet outil est fournie et commentée