SUPINFO Academic Dept.

SUPINFO International University

INSTITUTE OF INFORMATION TECHNOLOGY

# Group Projects
# A.Sc.1 - Architecture

**Contents**

2015-2016

Version 1.0
Last update: 8/12/2015
Use: Students/Staff
Author: Samuel CUELLA

# TABLE OF CONTENTS

# 1. Project Overview

Nephi Toys inc. sells various toys for young children. Their new product is a virtual piano that plugs on TV. Children can learn simple piano "songs" (pieces) just by following what's on screen.

Nephi will use PC-like hardware and wants you to write a prototype in x86 assembly. Your prototype will use the PC keyboard instead of a piano-like keyboard which will be shipped with the actual product.

# 2. Functional Expression

## 2.1. Operation

When booted, the device will show a standard piano keyboard on screen. All piano keys are mapped to a PC key. When keys are pressed on the physical keyboard, the correct sound is generated by the computer and played through the soundcard. Key press is also replicated on screen by re-drawing the pressed keys in a the "pressed" state. Users can play in three modes:

- Free to play (default)
- Training
- Watch me play

In "free to play" mode, the user can play which ever piece he wants. The screen show the pressed keys and sound is played accordingly.

In the training mode, the user can select a piece to work on (provide at least one with your prototype). The computer will then highlight the key(s) to press on screen : The user just have to press the according key on his keyboard in sequence.

In the watch me play mode, the user selects a piece to be played by the computer. The computer then plays the piece, showing the keys being pressed, just like if the user were actually pressing them.

## 2.2. Technical requirements

Your prototype must work on the bare metal: The machine should boot directly to your program. It must work on VMware/VirtualBox/Bochs, at your option. You can use BIOS routines to draw on screen. You can use the PC Speaker or directly drive the PCI soundcard emulated by VMware/VirtualBox/Bochs.

## 2.3. Bonus features

Create an actual physical piano keyboard that behave like a PC keyboard and sends correct byte sequences for each piano key. You could use scrap toys and old keyboards for that.

# 3. Deliverables

Students should include the following elements in their final delivery:

- A zip archive with the project source code. The source code must also come with the build system used (Project file, autotools...), if any.
- Project documentation, based on the template.
  - Technical documentation explaining your choices and/or implementation choices/details on the following items (at least):
    - Generating the sound tones
  - User manual (how to play)

**The first document is an academic document. Address the reader as a teacher, not a client. The last one (game manual) should address the reader as a user. These documents can be in French or in English, at your option.**

# 4. Graded Items

The project will be graded as follows, on a 26/20 scale:

- Base operation/Free to play (11 points)
  - The screen shows a piano-like keyboard (2.5 points)
  - PC keys are mapped to piano keys: Each note can be heard by pressing the key (6 points)
  - The key press is replicated on screen (2.5 points)
- Training mode (4 points)
  - Users can select pieces (at least one) (1 point)
  - The computer highlight the next note to be played (2 points)
  - The computer waits for the key to be pressed to continue (1 point)
- Watch me play (5 points)
  - Users can select piece(s) (1 point)
  - The computer plays the piece (2 points)
  - The computer shows each key pressed/released (2 points)
- Bonus (6 points)
  - Making an actual keyboard (3 points)
  - Any other bonus features done by the students (3 points)