

## Exercici 3 de laboratori d'IDI 2020-2021, Q2

### Instruccions

1. Aquests exercicis són individuals, així que només pots lliurar **codi que hakis generat tu**. No pots fer servir codi que altres estudiants hagin compartit amb tu (ni que tu hakis compartit amb d'altres estudiants). Altrament es considerarà còpia.
2. Per a fer el lliurament has de generar un arxiu tar que inclogui tot el codi del teu exercici i que es digui `<nom-usuari>-Ex3.tgz`, on substituiràs `<nom-usuari>` pel teu nom d'usuari. Fes que el directori de treball sigui aquell en el què has desenvolupat el codi de l'exercici i, si per exemple el teu nom és Pompeu Fabra, has d'executar

```
make distclean
tar zcvf pompeu.fabra-Ex3.tgz *
```

3. Un cop fet això, al teu directori tindràs l'arxiu `<nom-usuari>-Ex3.tgz` que és el que has de lliurar a la pràctica corresponent del Racó de la FIB **abans del diumenge dia 9 de maig a les 23:59**.

### Enunciat

En aquest exercici es demana que facis una aplicació en Qt per simular un àbac binari capaç de comptar valors octals fins a valors de dos dígit. **Important: La finestra de l'aplicació s'ha de titular *Abacus* (si no és així no us podem assegurar que es tingui en compte el lliurament)**. Els elements i funcionalitats en la interfície d'aquesta aplicació han de ser:

- L'àbac consistirà en **2 files de 3 etiquetes (QLabel)**. Una fila representa les decenes i l'altra les unitats (les decenes van dalt i les unitats baix). Cada etiqueta només podrà tenir valors 0 o 1, i si el valor és 0 el color de fons de l'etiqueta serà verd i si el valor és 1 el color serà groc. Cada fila (la de les decenes i la de les unitats) poden representar conjuntament (les tres etiquetes juntes) valors entre el 0 i el 7 (en binari). El conjunt de les 6 etiquetes representen el valor total en octal del nombre que es representa (pot anar de 0 a 77 en octal). Totes les etiquetes comencen amb valor 0 per defecte.
- El funcionament de l'Àbac es controlarà mitjançant **un botó d'increment** que sempre incrementa el valor total en 1 única unitat (la unitat està representada en l'etiqueta de baix a la dreta). Quan es prem aquest botó, l'àbac ha d'incrementar en 1 el valor que representa. Això vol dir que l'etiqueta de baix a la dreta ha de modificar el seu valor i si aquest valor ja era 1 (no pot créixer més) ha de passar a modificar el valor de l'etiqueta de la seva esquerra i ella passar a valor 0, i així successivament. Quan es fa increment sobre l'Àbac i aquest estava amb valor 77, ha de tornar al valor 0 inicial. Aquest botó ha de tenir un accelerador **Alt-i**.

Si posem un exemple numèric, suposem que tenim el nombre octal representat en binari següent:

```
0  1  0  (representa decenes = 2)
1  1  1  (representa unitats = 7)
```

Si en aquest punt fem *incrementa* clicant en el botó, la representació que hauríem d'obtenir és:

```
0  1  1  (representa decenes = 3)
0  0  0  (representa unitats = 0)
```

- Hi haurà també un `QLCDNumber` que representi en tot moment el nombre total de l'àbac en representació octal (quin nombre està donant l'àbac en cada moment). En l'exemple anterior aquest `QLCDNumber` tindrà primer un 27 i després un 30.
- La interfície haurà d'incloure també **un element més** que permeti en qualsevol moment tornar a l'inici tota l'aplicació. Aquest retorn a l'inici s'ha de poder fer també mitjançant **Alt-R**.

Cal que el disseny compleixi els principis de disseny vistos a classe i que la interfície es comporti bé quan es redimensioni.

**Important:** *Per implementar aquest exercici heu de derivar de les classes `QLabel` i `QLCDNumber`.*