

# P1: Gestió de l'entrada / sortida

Leonor Frias

September 12, 2007

## 1 Entrada/sortida manual

Fins ara hem vist, que els programes poden

- llegir dades d'entrada introduïdes per teclat: `cin >> var`
- escriure dades per la pantalla: `cout << var`

**Problemes :**

- d'introduir dades manualment:
  - no podem corregir línia un cop apretat enter
  - pesat per moltes dades d'entrada, cada cop que volem provar programa!
  - Ctrl+D per acabar entrada
- de que les dades sortin per la pantalla:

Podem no distingir si la sortida correspon exactament amb la sortida que s'espera el programa. Concretament, podem tenir problemes amb:

  - espais
  - línies en blanc
  - ...

Recordeu que el format ha de coincidir exactament per a que el programa es consideri correcte.

## 2 Automatització

1. Redireccionar entrada i/o sortida del programa a fitxers:

```
./prog.x < prog.ent > prog.sor
```

on concretament:

- `prog.ent`: La podem escriure nosaltres mateixos, o millor baixar-nos el fitxer `ex.ent` del jutge.  
Us recomano que us baixeu l'entrada del jutge quan aquesta sigui llarga.

- `prog.sor`: Fitxer que es genera amb la sortida del nostre programa (tot el que escrivim amb `cout`).

Noteu que si heu escrit pel canal `cerr`, aquesta part se us seguirà escrivint per pantalla.

2. Comparem la sortida amb el resultat que hauria de donar segons el joc de proves públic.

- (a) Donar primera ullada mirant contingut del fitxer. Per fer-ho usar:

```
more prog.sor  
less prog.sor
```

- (b) Si no veiem errors aparentment, comparar (automàticament) amb la sortida esperada del jutge. Per a poder comparar-ho, cal que la sortida esperada estigui tb en un fitxer: la podem escriure nosaltres mateixos, o millor baixar-nos el fitxer `ex.cor` del jutge.

```
cmp prog.sor prog.cor  
diff --side-by-side --left-column prog.sor prog.cor
```

Us recomano que us baixeu el fitxer de sortida sempre a menys que sigui obvi. A l'examen sobretot feu-ho!