

MC937A/MO603A – Computação Gráfica - 2021-S2 - Jorge Stolfi
Trabalho de laboratório 07 - 2021-10-06
Robôs com EA - O Filme

Objetivos: treinar os conceitos básicos de animação por computador.

Enunciado. Um filme com boa animação por computador pode ser melhor que a realidade, ao ponto de torná-la desnecessária. Por esse princípio, em vez de construir nosso robô alienígena, deve ser suficiente produzir um gif animado do mesmo. Mesmo um simples movimento repetitivo deve bastar para declararmos a guerra já vencida.

Execução. Produza uma macro `quadro(tf)` que devolve o robô, com as posições das juntas correspondentes a um instante arbitrário `tf` do filme; sendo `tf=0` e `tf=1` produzem a posição inicial, e valores intermediários de `tf` produzem outras posições intermediárias, com variação suave. Você vai precisar de uma macro `robo_interpola(tt, t0,t1, P0,P1)`, semelhante a `robo_mov` da aula passada que recebe dois vetores de parâmetros `P0` e `P1`, os tempos correspondentes `t0` e `t1`, e interpola um vetor `P` entre os mesmos, e chama `robo_vet` com esse vetor..

Arquivos. Copie os arquivos da aula passada para uma nova sub-pasta `2021-10-06` da pasta `mc937` no seu computador. Edite o arquivo `main.pov`, conforme solicitado acima. Execute o comando `make fast` numa shell para gerar a imagem, e use seu bráuzer (com `file:///home/.../main.gif`) para ver o gif animado.

Exportação. Não se esqueça de **exportar seu arquivo `main.pov` até as 21:00 para http://students.ic.unicamp.br/~raSEU_RA/mc937-2021-2/2021-10-06/**

Comandos. Os seguintes comandos de POV-Ray são relevantes para esta tarefa:

Movimento contínuo por partes. Um movimento cíclico suave não pode ser executado com uma única interpolação de parâmetros `interpola3` (muito menos `interpola1`). Então é necessário usar várias interpolações, como segue:

```
#macro quadro(tf)
  #local T0 = 0.00;
  #local P0 = ...
  #local T1 = 0.25;
  #local P1 = ...
  #local T2 = 0.50;
  #local P2 = ...
  #local T3 = 0.80;
  #local P3 = ...
  #local T4 = 1.00;
  #local P4 = P0
  #if ((T0 <= tf) & (tf <= T1))
    #local qd = robo_interpola(tf, T0,T1, P0,P1);
  #elseif ((T1 <= tf) & (tf <= T2))
    #local qd = robo_interpola(tf, T1,T2, P1,P2);
  #elseif ((T2 <= tf) & (tf <= T3))
```

```
    #local qd = robo_interpola(tf, T2,T3, P2,P3);  
#elseif ((T3 <= tf) & (tf <= T4))  
    #local qd = robo_interpola(tf, T3,T4, P3,P4);  
#end  
qd  
#end
```

Por enquanto, sugiro usar interpolação afim (`interpola1`) na macro `robo_interpola(tt, t0,t1, P0,P1)`. Em aula futura veremos como usar (`interpola3`) para obter um movimento mais suave.

Chamando a macro quadro. Use `object{ quadro(clock) }`.

Executando o make. Numa shell, execute `make fast`, `make movie`, ou `make CLOCK=0.1234 still` para gerar apenas um quadro.