



UNIVERSIDADE FEDERAL DE SANTA CATARINA

Lucas Machado da Palma	12200640.
Luiz Henrique Urias	12200647.

Trabalho Prolog T2A

– Base de Dados –

Para essa base de dados, algumas alterações foram necessárias, como o uso do dynamic, para o funcionamento do assert. E a criação de 6 novos arquivos, usados para a inserção de novos dados ao banco, são eles : [novasCidades](#), [novosCursos](#), [novosNasc](#), [novosNomes](#), [novosTelefones](#) e [novosXP](#).

```
:- dynamic(nome/1).
nome('Suzana Vieira').
nome('Mary Jane').
nome('Pablo Monte').
nome('Lucifer Petry').
nome('Bruce Wayne').
nome('Oliver Queen').
nome('Natasha Romanov').
nome('Logan Silva').
nome('Silva e Silva').
nome('Barry Allen').
```

```
:- dynamic(nasc/4).
nasc('Bruce Wayne', 22, 02, 1939).
nasc('Suzana Vieira', 10, 20, 1959).
nasc('Mary Jane', 15, 07, 1950).
nasc('Lucifer Petry', 06, 06, 1966).
nasc('Oliver Queen', 15, 10, 1979).
nasc('Natasha Romanov', 10, 06, 1978).
nasc('Logan Silva', 29, 12, 1969).
nasc('Silva e Silva', 22, 02, 1982).
nasc('Barry Allen', 22, 02, 1986).
nasc('Pablo Monte', 25, 11, 1982).
```

```
:- dynamic(professor/1).
professor('Joao Tenorio').
professor('Aline Vemcurt').
professor('Tereza Montegrande').
professor('Clorisvaldo de Aumeida').
professor('Pedro Pedrozo').
```

```
:- dynamic(orientou/2).
orientou('Joao Tenorio', 'Barry Allen').
orientou('Tereza Montegrande', 'Pablo Monte').
orientou('Tereza Montegrande', 'Suzana Vieira').
orientou('Clorisvaldo de Aumeida', 'Mary Jane').
orientou('Joao Tenorio', 'Lucifer Petry').
orientou('Joao Tenorio', 'Bruce Wayne').
orientou('Clorisvaldo de Aumeida', 'Oliver Queen').
orientou('Joao Tenorio', 'Natasha Romanov').
orientou('Pedro Pedrozo', 'Logan Silva').
orientou('Pedro Pedrozo', 'Silva e Silva').
```

```
:- dynamic(curso/1).
curso('Ciencias da Computacao').
```

curso('Letras').
curso('Numeros').
curso('Artes Marciais').
curso('Sistemas').
curso('Engenharia Mecanica').

:- dynamic(cursou/4).
cursou('Bruce Wayne', 'Artes Marciais', 1980, 1985).
cursou('Oliver Queen', 'Artes Marciais', 1982, 1987).
cursou('Suzana Vieira', 'Letras', 1982, 1987).
cursou('Mary Jane', 'Engenharia Mecanica', 1990, 2000).
cursou('Pablo Monte', 'Numeros', 1990, 1994).
cursou('Pablo Monte', 'Sistemas', 1995, 2000).
cursou('Lucifer Petry', 'Letras', 1966, 1972).
cursou('Natasha Romanov', 'Ciencias da Computacao', 1995, 2000).
cursou('Logan Silva', 'Artes Marciais', 1970, 1977).
cursou('Silva e Silva', 'Numeros', 2000, 2006).
cursou('Barry Allen', 'Sistemas', 1997, 2001).

:- dynamic(empresa/1).
empresa('LABSEC').
empresa('UFSC').
empresa('Google').
empresa('Microsoft').
empresa('Wayne Enterprises').
empresa('Queen Industries').
empresa('Samsung').

:- dynamic(telefone/1).
telefone(33333333).
telefone(44444444).
telefone(76547636).
telefone(87568444).
telefone(87468923).
telefone(98769867).
telefone(49654348).
telefone(87565439).
telefone(98766543).
telefone(35899864).

:- dynamic(cidade/1).
cidade('Florianopolis').
cidade('Gotham').
cidade('Starling').
cidade('NewYork').
cidade('Central').

:- dynamic(telefonede/2).
telefonede('Suzana Vieira', 33333333).
telefonede('Pablo Monte', 44444444).
telefonede('Bruce Wayne', 76547636).
telefonede('Natasha Romanov', 87468923).

telefonede('Barry Allen', 98769867).
telefonede('Silva e Silva', 49654348).
telefonede('Oliver Queen', 87468923).
telefonede('Lucifer Petry', 87565439).
telefonede('Mary Jane', 98766543).
telefonede('Logan Silva', 35899864).
:- dynamic(cidadede/2).
cidadede('Pablo Monte', 'Florianopolis').
cidadede('Logan Silva', 'Florianopolis').
cidadede('Suzana Vieira', 'Central').
cidadede('Bruce Wayne', 'Gotham').
cidadede('Natasha Romanov', 'NewYork').
cidadede('Barry Allen', 'Central').
cidadede('Silva e Silva', 'NewYork').
cidadede('Oliver Queen', 'Starling').
cidadede('Lucifer Petry', 'Florianopolis').
cidadede('Mary Jane', 'Central').

:- dynamic(trabalhou/2).
trabalhou('Pablo Monte', 'LABSEC').
trabalhou('Pablo Monte', 'Microsoft').
trabalhou('Pablo Monte', 'Google').
trabalhou('Logan Silva', 'Google').
trabalhou('Suzana Vieira', 'Google').
trabalhou('Bruce Wayne', 'Wayne Enterprises').
trabalhou('Natasha Romanov', 'Queen Industries').
trabalhou('Barry Allen', 'UFSC').
trabalhou('Silva e Silva', 'UFSC').
trabalhou('Oliver Queen', 'Queen Industries').
trabalhou('Lucifer Petry', 'LABSEC').
trabalhou('Mary Jane', 'Samsung').

:- dynamic(referencia/2).
referencia('Oliver Queen', 'Bruce Wayne').
referencia('Logan Silva', 'Bruce Wayne').
referencia('Silva e Silva', 'Pablo Monte').
referencia('Lucas Silva', 'Pablo Monte').
referencia('Mary Jane', 'Pablo Monte').
referencia('Silva e Silva', 'Lucifer Petry').
referencia('Mary Jane', 'Barry Allen').
referencia('Natasha Romanov', 'Barry Allen').
referencia('Natasha Romanov', 'Pablo Monte').
referencia('Natasha Romanov', 'Mary Jane').

:- dynamic(inst/1).
inst('UFSC').
inst('UDESC').
inst('UNISUL').
inst('UFSM').
inst('QueenSchool').
inst('Harvard').
inst('Vida').

```

:- dynamic(estudouEm/2).
estudouEm('Pablo Monte', 'UNISUL').
estudouEm('Logan Silva', 'UNISUL').
estudouEm('Suzana Vieira', 'UNISUL').
estudouEm('Suzana Vieira', 'UFSC').
estudouEm('Mary Jane', 'UDESC').
estudouEm('Lucifer Petry', 'UFSM').
estudouEm('Oliver Queen', 'QueenSchool').
estudouEm('Silva e Silva', 'Vida').
estudouEm('Barry Allen', 'Harvard').
estudouEm('Natasha Romanov', 'UFSC').

```

```

:- dynamic(xp/5).
xp('Pablo Monte', 'Facilitador', 'LABSEC', 2001, 2007).
xp('Pablo Monte', 'Testador', 'Microsoft', 2008, 2010).
xp('Pablo Monte', 'Gerente Organizacional dos Facilitadores', 'Google', 2011, 2014).
xp('Logan Silva', 'Programador', 'Google', 2000, 2009).
xp('Logan Silva', 'Fisiculturista', 'UFSC', 2010, 2015).
xp('Suzana Vieira', 'Secretaria', 'Google', 1980, 2012).
xp('Bruce Wayne', 'Bilionário', 'Wayne Enterprises', 2010, 2015).
xp('Natasha Romanov', 'Arquiteta de Projetos', 'Queen Industries', 1997, 2014).
xp('Barry Allen', 'Facilitador', 'UFSC', 2000, 2009).
xp('Silva e Silva', 'Facilitador', 'UFSC', 2001, 2008).
xp('Oliver Queen', 'CEO', 'Queen Industries', 2013, 2014).
xp('Lucifer Petry', 'Comentarista', 'LABSEC', 2005, 2007).
xp('Mary Jane', 'Diretora de Marketing', 'Microsoft', 1970, 2005).

```

```

listlen([ ], 0).
listlen([H|T], N) :- listlen(T, N1), N is N1 + 1.
conta(P, Contador):-findall(1, P, Lista), listlen(Lista, Contador).
acumula([ ], 0).
acumula([H|T], N):-acumula(T, N1), N is N1 + H.
acumulaMedia([ ], 0).
acumulaMedia([H|T], K):-acumulaMedia(T, K1), totalEstudado(H,L), K is K1 + L.
max([X],X).
max([X|Xs],X) :- max(Xs,Y), X >= Y.
max([X|Xs],N) :- max(Xs,N), N > X.
inserir(Elem, Lista, [Elem|Lista]).

```

– Questões –

Uma observação importante é sobre os passos necessários para a compilação e execução do código criado. Encontramos algumas falhas e barreiras com o uso de assert e dynamic no *gprolog*. Por esses motivos a ferramenta que se mostrou mais eficiente foi o *swi-prolog*.

Para chama-lo são necessários os seguintes passos no terminal:

```
swipl -o t2.pl, novasCidades.pl, novosCursou.pl, novosNasc.pl, novosNomes.pl, novosTelefones.pl, novosXP.pl
```

e

```
[t2, novasCidades, novosCursou, novosNasc, novosNomes, novosTelefones, novosXP].
```

A seguinte tele deverá aparecer, informando que todos os arquivos pl requeridos foram compilados. Uma vez que alterações foram feitas nos arquivos pl, mensagens de warning poderão aparecer sendo necessário executar o segundo comando em azul novamente.

```
?- [t2, novasCidades, novosCursou, novosNasc, novosNomes, novosTelefones, novosXP].
Warning: /home/12200640/t2.pl:182:
Singleton variables: [H]
Warning: /home/12200640/t2.pl:215:
Singleton variables: [Z]
Warning: /home/12200640/t2.pl:215:
Singleton variable in \+: Z
Warning: /home/12200640/t2.pl:218:
Singleton variables: [Y]
Warning: /home/12200640/t2.pl:221:
Singleton variables: [W]
Warning: /home/12200640/t2.pl:233:
Singleton variables: [A]
Warning: /home/12200640/t2.pl:257:
Singleton variables: [X,I,F]
Warning: /home/12200640/t2.pl:275:
Singleton variables: [X]
Warning: /home/12200640/t2.pl:280:
Singleton variables: [Z]
Warning: /home/12200640/t2.pl:281:
Singleton variables: [X]
Warning: /home/12200640/t2.pl:285:
Singleton variables: [X]
Warning: /home/12200640/t2.pl:289:
Singleton variables: [X]
Warning: /home/12200640/t2.pl:317:
Singleton variables: [C]
% t2 compiled 0.01 sec, 229 clauses
% novasCidades compiled 0.00 sec, 1 clauses
% novosCursou compiled 0.00 sec, 1 clauses
% novosNasc compiled 0.00 sec, 1 clauses
% novosNomes compiled 0.00 sec, 1 clauses
% novosTelefones compiled 0.00 sec, 1 clauses
% novosXP compiled 0.00 sec, 1 clauses
true.
?- █
```

Todas as perguntas feitas neste trabalho estão acompanhadas de exemplos de respostas em imagens, no caso de alterações na base de dados como as inclusões, alterações e remoções, foram anexadas também imagens da implicação da meta nos respectivos arquivos pl.

Inclusão de dados pelo nome.

Como foi acordado com o professor previamente, foram escolhidas apenas algumas metas da base de dados que poderiam ser inseridas, não havendo a necessidade de cobrir todas as possibilidades. As escolhidas foram: nome, experiência (xp), nascimento (nasc), telefone, cursou e cidade.

1)

gravarNome:-tell('novosNomes.pl'), listing(nome), told.

inserirNome(Nome):-assert(nome(Nome)), gravarNome, !.

```
?- inserirNome('Pessoa Nova').  
true.  
  
?-
```

```
:- dynamic nome/1.  
  
nome('Suzana Vieira').  
nome('Mary Jane').  
nome('Pablo Monte').  
nome('Lucifer Petry').  
nome('Bruce Wayne').  
nome('Oliver Queen').  
nome('Natasha Romanov').  
nome('Logan Silva').  
nome('Silva e Silva').  
nome('Barry Allen').  
nome('Pessoa Nova').
```

2)

gravarXP:-tell('novoXP'), listing(xp), told.

inserirXP(NOME, CARGO, EMPRESA, INICIO, FIM):-assert(xp(NOME, CARGO, EMPRESA, INICIO, FIM)), gravarXP.

```
?- inserirXP('Pessoa Nova', 'Trabalhador', 'LABSEC', 2000, 2005).  
true.  
  
?-
```

```
:- dynamic xp/5.  
  
xp('Pablo Monte', 'Facilitador', 'LABSEC', 2001, 2007).  
xp('Pablo Monte', 'Testador', 'Microsoft', 2008, 2010).  
xp('Pablo Monte', 'Gerente Organizacional dos Facilitadores', 'Google', 2011, 2014).  
xp('Logan Silva', 'Programador', 'Google', 2000, 2009).  
xp('Logan Silva', 'Fisiculturista', 'UFSC', 2010, 2015).  
xp('Suzana Vieira', 'Secretaria', 'Google', 1980, 2012).  
xp('Bruce Wayne', 'Bilionário', 'Wayne Enterprises', 2010, 2015).  
xp('Natasha Romanov', 'Arquiteta de Projetos', 'Queen Industries', 1997, 2014).  
xp('Barry Allen', 'Facilitador', 'UFSC', 2000, 2009).  
xp('Silva e Silva', 'Facilitador', 'UFSC', 2001, 2008).  
xp('Oliver Queen', 'CEO', 'Queen Industries', 2013, 2014).  
xp('Lucifer Petry', 'Comentarista', 'LABSEC', 2005, 2007).  
xp('Mary Jane', 'Diretora de Marketing', 'Microsoft', 1970, 2005).  
xp('Pessoa Nova', 'Trabalhador', 'LABSEC', 2000, 2005).
```

3)

gravarNasc:-tell('novosNasc'), listing(nasc), told.

inserirNasc(Nome, DIA, MES, ANO):-assert(nasc(Nome, DIA, MES, ANO)), gravarNasc.

```
?- inserirNasc('Pessoa Nova', 03, 03, 1993).  
true.  
?-
```

```
:- dynamic nasc/4.  
  
nasc('Bruce Wayne', 22, 2, 1939).  
nasc('Suzana Vieira', 10, 20, 1959).  
nasc('Mary Jane', 15, 7, 1950).  
nasc('Lucifer Petry', 6, 6, 1966).  
nasc('Oliver Queen', 15, 10, 1979).  
nasc('Natasha Romanov', 10, 6, 1978).  
nasc('Logan Silva', 29, 12, 1969).  
nasc('Silva e Silva', 22, 2, 1982).  
nasc('Barry Allen', 22, 2, 1986).  
nasc('Pablo Monte', 25, 11, 1982).  
nasc('Pessoa Nova', 3, 3, 1993).
```

4)

gravarTelefone:- tell('novosTelefones.pl'),listing(telefonede), told.

inserirTelefone(Nome, TEL):- nome(Nome), assert(telefonede(Nome,TEL)), gravarTelefone, !.

```
?- inserirTelefone('Pessoa Nova', 33333333).  
true.  
?-
```

```
:- dynamic telefonede/2.  
  
telefonede('Suzana Vieira', 33333333).  
telefonede('Pablo Monte', 44444444).  
telefonede('Bruce Wayne', 76547636).  
telefonede('Natasha Romanov', 87468923).  
telefonede('Barry Allen', 98769867).  
telefonede('Silva e Silva', 49654348).  
telefonede('Oliver Queen', 87468923).  
telefonede('Lucifer Petry', 87565439).  
telefonede('Mary Jane', 98766543).  
telefonede('Logan Silva', 35899864).  
telefonede('Pessoa Nova', 33333333).
```


5)

gravarCurso:- tell('novosCursou.pl'), listing(cursou), told.

*inserirCurso(Nome,Curso,Inicio,Fim):- nome(Nome), assert(cursou(Nome,Curso,Inicio,Fim)),
gravarCurso,!.*

```
?- inserirCurso('Pessoa Nova', 'CC0', 2010, 2014).  
true.  
?-
```

```
:- dynamic cursou/4.  
  
cursou('Bruce Wayne', 'Artes Marciais', 1980, 1985).  
cursou('Oliver Queen', 'Artes Marciais', 1982, 1987).  
cursou('Suzana Vieira', 'Letras', 1982, 1987).  
cursou('Mary Jane', 'Engenharia Mecanica', 1990, 2000).  
cursou('Pablo Monte', 'Numeros', 1990, 1994).  
cursou('Pablo Monte', 'Sistemas', 1995, 2000).  
cursou('Lucifer Petry', 'Letras', 1966, 1972).  
cursou('Natasha Romanov', 'Ciencias da Computacao', 1995, 2000).  
cursou('Logan Silva', 'Artes Marciais', 1970, 1977).  
cursou('Silva e Silva', 'Numeros', 2000, 2006).  
cursou('Barry Allen', 'Sistemas', 1997, 2001).  
cursou('Pessoa Nova', 'CC0', 2010, 2014).
```

6)

gravarCidade:-tell('novasCidades.pl'), listing(cidadede, cidade), told.

*inserirCidade(NOME,CIDADE):-
nome(NOME),cidade(CIDADE),assert(cidadede(NOME,CIDADE)), gravarCidade,!.*

```
?- inserirCidade('Pessoa Nova', 'Florianopolis').  
true.  
?-
```

```
:- dynamic cidadede/2.  
  
cidadede('Pablo Monte', 'Florianopolis').  
cidadede('Logan Silva', 'Florianopolis').  
cidadede('Suzana Vieira', 'Central').  
cidadede('Bruce Wayne', 'Gotham').  
cidadede('Natasha Romanov', 'NewYork').  
cidadede('Barry Allen', 'Central').  
cidadede('Silva e Silva', 'NewYork').  
cidadede('Oliver Queen', 'Starling').  
cidadede('Lucifer Petry', 'Florianopolis').  
cidadede('Mary Jane', 'Central').  
cidadede('Pessoa Nova', 'Florianopolis').
```

Alteração de dados (pode ser apenas da cidade e do telefone).

1)

imprimirCidade(NOME):- write('Nome: '), write(NOME), nl, write('Cidade: '), cidadede(NOME, U), write(U).

alterarCidade(NOME):-nome(NOME), imprimirCidade(NOME), nl, removerCidade(NOME), write('Nova Cidade\n'), read(NovaCidade), assert(cidadede(NOME, NovaCidade)), gravarCidade,!.

```
?- alterarCidade('Pessoa Nova').
Nome: Pessoa Nova
Cidade: Florianopolis
Nova Cidade
|: 'Gotham'.
true.

?- 
```

```
:- dynamic cidadede/2.

cidadede('Pablo Monte', 'Florianopolis').
cidadede('Logan Silva', 'Florianopolis').
cidadede('Suzana Vieira', 'Central').
cidadede('Bruce Wayne', 'Gotham').
cidadede('Natasha Romanov', 'NewYork').
cidadede('Barry Allen', 'Central').
cidadede('Silva e Silva', 'NewYork').
cidadede('Oliver Queen', 'Starling').
cidadede('Lucifer Petry', 'Florianopolis').
cidadede('Mary Jane', 'Central').
cidadede('Pessoa Nova', 'Gotham').
```

2)

imprimirTelefone(NOME):-write('Nome: '), write(NOME), nl, write('Telefone: '), telefonede(NOME,U), write(U).

alterarTelefone(NOME):-nome(NOME), imprimirTelefone(NOME), nl, removerTelefone(NOME), write('Novo Telefone\n'), read(NovoTelefone), assert(telefonede(NOME, NovoTelefone)), gravarTelefone,!.

```
?- alterarTelefone('Pessoa Nova').
Nome: Pessoa Nova
Telefone: 44444444
Novo Telefone
|: 99999999.
true.

?- 
```

```
?- dynamic telefonede/2.

telefonede('Suzana Vieira', 33333333).
telefonede('Pablo Monte', 44444444).
telefonede('Bruce Wayne', 76547636).
telefonede('Natasha Romanov', 87468923).
telefonede('Barry Allen', 98769867).
telefonede('Silva e Silva', 49654348).
telefonede('Oliver Queen', 87468923).
telefonede('Lucifer Petry', 87565439).
telefonede('Mary Jane', 98766543).
telefonede('Logan Silva', 35899864).
telefonede('Pessoa Nova', 99999999).
```

Exclusão de dados pelo nome.

Da mesma forma que a inserção, ao deletar metas também foram escolhidas apenas algumas possibilidades.

1)

removerTelefone(NOME):-retract(telefonede(NOME,T)), gravarTelefone.

```
?- removerTelefone('Pessoa Nova').
true.

?- 
```

```
?- dynamic telefonede/2.

telefonede('Suzana Vieira', 33333333).
telefonede('Bruce Wayne', 76547636).
telefonede('Natasha Romanov', 87468923).
telefonede('Barry Allen', 98769867).
telefonede('Silva e Silva', 49654348).
telefonede('Oliver Queen', 87468923).
telefonede('Lucifer Petry', 87565439).
telefonede('Mary Jane', 98766543).
telefonede('Logan Silva', 35899864).
```

2)

removerCurso(NOME, CURSO):-retract(cursou(NOME,CURSO,_,_)), gravarCurso. .

```
?- removerCurso('Pessoa Nova', 'CC0').
true.

?- 
```

```
:- dynamic cursou/4.
```

```
cursou('Bruce Wayne', 'Artes Marciais', 1980, 1985).  
cursou('Oliver Queen', 'Artes Marciais', 1982, 1987).  
cursou('Suzana Vieira', 'Letras', 1982, 1987).  
cursou('Mary Jane', 'Engenharia Mecanica', 1990, 2000).  
cursou('Pablo Monte', 'Numeros', 1990, 1994).  
cursou('Pablo Monte', 'Sistemas', 1995, 2000).  
cursou('Lucifer Petry', 'Letras', 1966, 1972).  
cursou('Natasha Romanov', 'Ciencias da Computacao', 1995, 2000).  
cursou('Logan Silva', 'Artes Marciais', 1970, 1977).  
cursou('Silva e Silva', 'Numeros', 2000, 2006).  
cursou('Barry Allen', 'Sistemas', 1997, 2001).
```

3)

removerNasc(NOME):-retract(nasc(NOME,_,_,_)), gravarNasc.

```
?- removerNasc('Pessoa Nova').  
true.
```

```
?- 
```

```
:- dynamic nasc/4.
```

```
nasc('Bruce Wayne', 22, 2, 1939).  
nasc('Suzana Vieira', 10, 20, 1959).  
nasc('Mary Jane', 15, 7, 1950).  
nasc('Lucifer Petry', 6, 6, 1966).  
nasc('Oliver Queen', 15, 10, 1979).  
nasc('Natasha Romanov', 10, 6, 1978).  
nasc('Logan Silva', 29, 12, 1969).  
nasc('Silva e Silva', 22, 2, 1982).  
nasc('Barry Allen', 22, 2, 1986).  
nasc('Pablo Monte', 25, 11, 1982).
```

4)

removerCidade(NOME):-retract(cidadede(NOME, C)), gravarCidade.

```
?- removerCidade('Pessoa Nova').  
true.
```

```
?- 
```

```
:- dynamic cidadede/2.
```

```
cidadede('Pablo Monte', 'Florianopolis').  
cidadede('Logan Silva', 'Florianopolis').  
cidadede('Suzana Vieira', 'Central').  
cidadede('Bruce Wayne', 'Gotham').  
cidadede('Natasha Romanov', 'NewYork').  
cidadede('Barry Allen', 'Central').  
cidadede('Silva e Silva', 'NewYork').  
cidadede('Oliver Queen', 'Starling').  
cidadede('Lucifer Petry', 'Florianopolis').  
cidadede('Mary Jane', 'Central').  
cidadede('Pessoa Nova', 'Gotham').
```

Consultas de formas variadas para obtenção de um ou mais Cvs.

```
imprimirBorda:-write('-----'), nl.  
imprimirNasc(NOME):-nl, write('Data de Nascimento: '), nasc(NOME, DIA, MES, ANO),  
write(DIA), write('/'), write(MES), write('/'), write(ANO).  
imprimirXP(NOME):-nl, write('Experiencias: '), nl, xp(NOME, CARGO, EMPRESA, INICIO,  
FIM), write('Cargo: '), write(CARGO), nl, write('Empresa: '), write(EMPRESA), nl,  
write('Trabalhou de '),  
write(INICIO), write(' até '), write(FIM), nl.  
imprimirCid(NOME):-nl, cidadede(NOME, CIDADE), write('Cidade: '), write(CIDADE), nl.  
imprimirNome(NOME):-nl, write('Nome: '), write(NOME), nl.  
imprimirFormacao(NOME):-nl, cursou(NOME,CURSO,INICIO,FIM), write('Cursou: '),  
write(CURSO), write(' de '), write(INICIO), write(' a '), write(FIM).  
imprimirTel(NOME):-nl, write('Contato: '), telefonede(NOME, TEL), write(TEL), nl.
```

Consultas pelo nome.

```
imprimirCurriculo(NOME):-nl, imprimirBorda, imprimirNome(NOME), imprimirNasc(NOME),  
imprimirCid(NOME), imprimirTel(NOME), imprimirFormacao(NOME), nl, imprimirXP(NOME),  
nl, imprimirBorda.
```

```
?- imprimirCurriculo('Pablo Monte').  
  
-----  
  
Nome: Pablo Monte  
  
Data de Nascimento: 25/11/1982  
Cidade: Florianopolis  
  
Contato: 44444444  
  
Cursou: Numeros de 1990 a 1994  
  
Experiencias:  
Cargo: Facilitador  
Empresa: LABSEC  
Trabalhou de 2001 até 2007  
  
-----  
true ;  
Cargo: Testador  
Empresa: Microsoft  
Trabalhou de 2008 até 2010  
  
-----  
true ;  
Cargo: Gerente Organizacional dos Facilitadores  
Empresa: Google  
Trabalhou de 2011 até 2014  
  
-----  
true ;  
Cursou: Sistemas de 1995 a 2000  
  
Experiencias:  
Cargo: Facilitador  
Empresa: LABSEC  
Trabalhou de 2001 até 2007  
  
-----  
true ;  
Cargo: Testador  
Empresa: Microsoft  
Trabalhou de 2008 até 2010  
  
-----  
true ;  
Cargo: Gerente Organizacional dos Facilitadores  
Empresa: Google  
Trabalhou de 2011 até 2014  
  
-----  
true.  
  
?-
```

OBS: Em conversa em sala com o professor, ficou claro que o fato de ser necessário apertar “;” para que os resultados sejam exibidos na tela e a repetição de algumas cláusulas não seriam um problema para o resultado final do trabalho.

Consultas pela Cidade.

imprimirCurriculoCidade(CIDADE):-cidade(CIDADE), cidade(NOME,CIDADE), nl,
imprimirBorda, imprimirNome(NOME),
imprimirNasc(NOME), imprimirCid(NOME), imprimirTel(NOME), imprimirFormacao(NOME),
nl, imprimirXP(NOME), nl, imprimirBorda.

```
?- imprimirCurriculoCidade('Central').
```

```
-----  
Nome: Suzana Vieira
```

```
Data de Nascimento: 10/20/1959
```

```
Cidade: Central
```

```
Contato: 33333333
```

```
Cursou: Letras de 1982 a 1987
```

```
Experiencias:
```

```
Cargo: Secretaria
```

```
Empresa: Google
```

```
Trabalhou de 1980 até 2012
```

```
-----  
true ;  
-----
```

```
Nome: Barry Allen
```

```
Data de Nascimento: 22/2/1986
```

```
Cidade: Central
```

```
Contato: 98769867
```

```
Cursou: Sistemas de 1997 a 2001
```

```
Experiencias:
```

```
Cargo: Facilitador
```

```
Empresa: UFSC
```

```
Trabalhou de 2000 até 2009
```

```
-----  
true ;  
-----
```

```
Nome: Mary Jane
```

```
Data de Nascimento: 15/7/1950
```

```
Cidade: Central
```

```
Contato: 98766543
```

```
Cursou: Engenharia Mecanica de 1990 a 2000
```

```
Experiencias:
```

```
Cargo: Diretora de Marketing
```

```
Empresa: Microsoft
```

```
Trabalhou de 1970 até 2005
```

```
-----  
true.
```

Consultas pela Instituição.

imprimirCurriculoInst(INST):-inst(INST), estudouEm(NOME,INST), nl, imprimirBorda,
imprimirNome(NOME),
imprimirNasc(NOME), imprimirCid(NOME), imprimirTel(NOME), imprimirFormacao(NOME),
nl, imprimirXP(NOME), nl, imprimirBorda.

```
?- imprimirCurriculoInst('UFSC').
```

```
-----  
Nome: Suzana Vieira
```

```
Data de Nascimento: 10/20/1959  
Cidade: Central
```

```
Contato: 33333333
```

```
Cursou: Letras de 1982 a 1987
```

```
Experiencias:
```

```
Cargo: Secretaria
```

```
Empresa: Google
```

```
Trabalhou de 1980 até 2012
```

```
-----  
true ;  
-----
```

```
Nome: Natasha Romanov
```

```
Data de Nascimento: 10/6/1978  
Cidade: NewYork
```

```
Contato: 87468923
```

```
Cursou: Ciencias da Computacao de 1995 a 2000
```

```
Experiencias:
```

```
Cargo: Arquiteta de Projetos
```

```
Empresa: Queen Industries
```

```
Trabalhou de 1997 até 2014
```

```
-----  
true.  
-----
```

```
?- █
```

Consultas pela Empresa.

imprimirCurriculoEmpresa(EMPRESA):-empresa(EMPRESA), trabalhou(NOME,EMPRESA), nl,
imprimirBorda, imprimirNome(NOME), imprimirNasc(NOME),
imprimirCid(NOME), imprimirTel(NOME), imprimirFormacao(NOME), nl, imprimirXP(NOME),
nl, imprimirBorda.

```
?- imprimirCurriculoEmpresa('Queen Industries').
```

```
-----  
Nome: Natasha Romanov
```

```
Data de Nascimento: 10/6/1978
```

```
Cidade: NewYork
```

```
Contato: 87468923
```

```
Cursou: Ciencias da Computucao de 1995 a 2000
```

```
Experiencias:
```

```
Cargo: Arquiteta de Projetos
```

```
Empresa: Queen Industries
```

```
Trabalhou de 1997 até 2014
```

```
-----  
true ;
```

```
-----  
Nome: Oliver Queen
```

```
Data de Nascimento: 15/10/1979
```

```
Cidade: Starling
```

```
Contato: 87468923
```

```
Cursou: Artes Marciais de 1982 a 1987
```

```
Experiencias:
```

```
Cargo: CEO
```

```
Empresa: Queen Industries
```

```
Trabalhou de 2013 até 2014
```

```
-----  
true.
```

```
?- █
```


Consulta de pessoas de Florianópolis com mais de 20 anos.

fromFlorianopolis(NOME):- cidade(NOME,CIDADE), CIDADE='Florianopolis'.

maiorDe20(NOME):-idade(NOME, VALOR), VALOR>=20.

consultaDupla:-fromFlorianopolis(NOME), maiorDe20(NOME), imprimirCurriculo(NOME).

```
?- consultaDupla.
```

```
-----  
Nome: Logan Silva
```

```
Data de Nascimento: 29/12/1969
```

```
Cidade: Florianopolis
```

```
Contato: 35899864
```

```
Cursou: Artes Marciais de 1970 a 1977
```

```
Experiencias:
```

```
Cargo: Programador
```

```
Empresa: Google
```

```
Trabalhou de 2000 até 2009
```

```
-----  
true ;
```

```
Cargo: Fisiculturista
```

```
Empresa: UFSC
```

```
Trabalhou de 2010 até 2015
```

```
-----  
true ;  
-----
```

```
Nome: Lucifer Petry
```

```
Data de Nascimento: 6/6/1966
```

```
Cidade: Florianopolis
```

```
Contato: 87565439
```

```
Cursou: Letras de 1966 a 1972
```

```
Experiencias:
```

```
Cargo: Comentarista
```

```
Empresa: LABSEC
```

```
Trabalhou de 2005 até 2007
```

```
-----  
true ;  
-----
```

true ;

Nome: Pablo Monte

Data de Nascimento: 25/11/1982

Cidade: Florianopolis

Contato: 44444444

Cursou: Numeros de 1990 a 1994

Experiencias:

Cargo: Facilitador

Empresa: LABSEC

Trabalhou de 2001 até 2007

true ;

Cargo: Testador

Empresa: Microsoft

Trabalhou de 2008 até 2010

true ;

Cargo: Gerente Organizacional dos Facilitadores

Empresa: Google

Trabalhou de 2011 até 2014

true ;

Cursou: Sistemas de 1995 a 2000

Experiencias:

Cargo: Facilitador

Empresa: LABSEC

Trabalhou de 2001 até 2007

true ;

Cargo: Testador

Empresa: Microsoft

Trabalhou de 2008 até 2010

true ;

Cargo: Gerente Organizacional dos Facilitadores

Empresa: Google

Trabalhou de 2011 até 2014

true.

?- █