
```
% HENG LOW WEE
```

```
% U096901R
```

```
% Problem Set 4 Problem 3
```

```
:- op(1099,yf,;).
```

```
:- op(960,fx,if).
```

```
:- op(959,xfx,then).
```

```
:- op(958,xfx,else).
```

```
:- op(960,fx,while).
```

```
:- op(960,fx,for).
```

```
:- op(959,xfx,do).
```

```
compileExpr(K,E,E,T,T) :-
```

```
    integer(K),!,
```

```
    write('    esp -= 4 ; *(int*)&M[esp] = '),
```

```
    write(K),write(' ; // push '), writeln(K).
```

```
compileExpr(V,Ein,Eout,Tin,Tout) :-
```

```
    atom(V),!,
```

```
    ( member((V->Addr),Ein)
```

```
    -> Tout = Tin, Eout = Ein
```

```
    ; Tout is Tin+4, Eout = [(V->Tin)|Ein], Addr = Tin),
```

```
    write('    ecx = *(int*)&M['),
```

```
    write(Addr),
```

```
    write('] ; esp -= 4 ; *(int*)&M[esp] = ecx ; // push '),
```

```
    writeln(V).
```

```
compileExpr(Exp,Ein,Eout,Tin,Tout) :-
```

```
    Exp =.. [0,A,B],
```

```
    compileExpr(A,Ein,Eaux,Tin,Taux),
```

```
    compileExpr(B,Eaux,Eout,Taux,Tout),
```

```
    writeln('    ecx = *(int*)&M[esp] ; esp += 4 ;'),
```

```
    writeln('    eax = *(int*)&M[esp] ; esp += 4 ;'),
```

```
    write('    eax '), write(0), writeln('= ecx ;'),
```

```
    write('    esp -= 4 ; *(int*)&M[esp] = eax ; // push result of '),
```

```
    writeln(0).
```

```
% SNAKES ON THE PLANE
```

```
compile((V1,V2)=(E1,E2),Ein,Eout,Tin,Tout,L,L) :-
```

```
    compileExpr(E1,Ein,Eaux,Tin,Taux),
```

```
    writeln('    ebx = *(int*)&M[esp] ;'),
```

```
    compileExpr(E2,Eaux,Eaux1,Taux,Taux1),
```

```
    writeln('    edx = *(int*)&M[esp] ;'),
```

```
    ( member((V1->Addr), Eaux)
```

```
    -> Taux1 = Taux, Eaux1 = Eaux
```

```
    ; Taux1 is Taux+4, Eaux1 = [(V1->Taux)|Eaux], Addr = Taux),
```

```

writeln('    ecx = ebx ;'),
write('    *(int*)&M['),write(Addr),write('] = ecx ; // pop '),
writeln(V1),

( member((V2->Addr1), Eaux1)
-> Tout = Taux1, Eout = Eaux1
; Tout is Taux1+4, Eout = [(V2->Taux1)|Eaux1], Addr1 = Taux1),

writeln('    ecx = edx ;'),
write('    *(int*)&M['),write(Addr1),write('] = ecx ; // pop '),
writeln(V2),
!.
```

```

compile(V=E,Ein,Eout,Tin,Tout,L,L) :-
  compileExpr(E,Ein,Eaux,Tin,Taux),
  ( member((V->Addr),Eaux)
  -> Tout = Taux, Eout = Eaux
  ; Tout is Taux+4, Eout = [(V->Taux)|Eaux], Addr = Taux),
  writeln('    ecx = *(int*)&M[esp] ; esp += 4 ;'),
  write('    *(int*)&M['),write(Addr),write('] = ecx ; // pop '),
  writeln(V).

compile(if B then S1 else S2,Ein,Eout,Tin,Tout,Lin,Lout) :- !,
  B =.. [0,X,Y], La1 is Lin+1,
  ( 0 == (\=) -> Otrans = '!=' ; Otrans = 0 ),
  writeln('    // start of if-then-else statement'),
  compileExpr(X,Ein,Ea1,Tin,Ta1),
  compileExpr(Y,Ea1,Ea2,Ta1,Ta2),
  writeln('    ecx = *(int*)&M[esp] ; esp += 4 ;') ,
  writeln('    eax = *(int*)&M[esp] ; esp += 4 ;') ,
  write('    if ( eax '), write(Otrans),
  write(' ecx ) goto Lthen'), write(Lin), writeln('; // if condition'),
  compile(S2,Ea2,Ea3,Ta2,Ta3,La1,La2),
  write('    goto Lendif'),write(Lin),writeln(';'),
  write('Lthen'),write(Lin),writeln(':'),
  compile(S1,Ea3,Eout,Ta3,Tout,La2,Lout),
  write('Lendif'),write(Lin),writeln(':').

compile(if B then S,Ein,Eout,Tin,Tout,Lin,Lout) :- !,
  B =.. [0,X,Y], La1 is Lin+1,
  ( 0 == (\=) -> Otrans = '!=' ; Otrans = 0 ),
  writeln('    // start of if-then statement'),
  compileExpr(X,Ein,Ea1,Tin,Ta1),
  compileExpr(Y,Ea1,Ea2,Ta1,Ta2),
  writeln('    ecx = *(int*)&M[esp] ; esp += 4 ;') ,
  writeln('    eax = *(int*)&M[esp] ; esp += 4 ;') ,
```

```

    write('    if ( eax '), write(0trans),
    write(' ecx ) goto Lthen'), write(Lin), writeln('; // if condition'),
    write('    goto Lendif'),write(Lin),writeln(';'),
    write('Lthen'),write(Lin),writeln(':'),
    compile(S,Ea2,Eout,Ta2,Tout,La1,Lout),
    write('Lendif'),write(Lin),writeln(':').
compile(while B do S,Ein,Eout,Tin,Tout,Lin,Lout) :- !,
    B =.. [0,X,Y], La1 is Lin+1,
    (    0 == (\=) -> 0trans = '!=' ; 0trans = 0 ),
    write('Lwhile'),write(Lin),writeln(':'),
    compileExpr(X,Ein,Ea1,Tin,Ta1),
    compileExpr(Y,Ea1,Ea2,Ta1,Ta2),
    writeln('    ecx = *(int*)&M[esp] ; esp += 4 ;') ,
    writeln('    eax = *(int*)&M[esp] ; esp += 4 ;') ,
    write('    if ( eax '), write(0trans),
    write(' ecx ) goto Lwhilebody'), write(Lin), writeln(';'),
    write('    goto Lendwhile'),write(Lin),writeln(';'),
    write('Lwhilebody'),write(Lin),writeln(':'),
    compile(S,Ea2,Eout,Ta2,Tout,La1,Lout),
    write('    goto Lwhile'),write(Lin),writeln(';'),
    write('Lendwhile'),write(Lin),writeln(':').

% You spin my head right round right round
compile(for (S1;S2;S3) do S4,Ein,Eout,Tin,Tout,Lin,Lout) :- !,
    compile(S1; while S2 do {S4;S3},Ein,Eout,Tin,Tout,Lin,Lout).

compile(S1;S2,Ein,Eout,Tin,Tout,Lin,Lout) :- !,
    compile(S1,Ein,Eaux,Tin,Taux,Lin,Laux),
    compile(S2,Eaux,Eout,Taux,Tout,Laux,Lout).
compile(S;;Ein,Eout,Tin,Tout,Lin,Lout) :- !,
    compile(S,Ein,Eout,Tin,Tout,Lin,Lout).
compile({S},Ein,Eout,Tin,Tout,Lin,Lout) :- !,
    compile(S,Ein,Eout,Tin,Tout,Lin,Lout).

compileProg(P) :-
    writeln('#include <stdio.h>'),
    writeln('int eax,ebx,ecx,edx,esi,edi,ebp,esp;'),
    writeln('unsigned char M[10000];'),
    writeln('void exec(void) {'),
    compile(P,[],Eout,0,_,0,_),
    writeln('}}}') ,nl,
    writeln('int main() {'),
    writeln('    esp = 10000 ;'),
    writeln('    exec();'),
    outputVars(Eout),

```

```
writeln('    return 0;'),  
writeln('}').
```

```
outputVars([]).
```

```
outputVars([(V->Addr)|T]) :-
```

```
    write('    printf("'),write(V),write('=%d\\n",'),  
    write('* (int*)&M['),write(Addr),writeln(']);'),  
    outputVars(T).
```

```
:- P = (
```

```
    s = 0 ;
```

```
    for ( i = 0 ; i < 10 ; i = i + 1 ) do {
```

```
        s = s + i ;
```

```
    } ;
```

```
),
```

```
compileProg(P).
```