

ADAS Subroutine qlpr

```
FUNCTION QLPR (Z1, N1, N2, E1, ZP, ATMSSP)
```

```
C
```

```
IMPLICIT REAL*8 (A-H, O-Z)
```

```
C
```

```
C-----
```

```
C
```

```
C ***** FORTRAN77 FUNCTION: QLPR *****
```

```
C
```

```
C-----
```

```
C PURPOSE: CALCULATE LODGE-PERCIVAL-RICHARDS ION IMPACT EXCITATION  
C CROSS-SECTIONS IN ORIGINAL FORM (J.PHYS.B. (1976)9,239).
```

```
C
```

```
C EXCITATION CROSS-SECTION IS EVALUATED AND DE-EXCITATION CROSS-SECTION  
C OBTAINED BY DETAILED BALANCE
```

```
C
```

```
C SCALING TO ARBITRARY PROJECTILE CHARGE FOLLOWS RECOMMENDATIONS  
C OF RIENHOLD,, OLSEN & FRITSCH (1990)PHYS.REV.A 41,4837
```

```
C
```

```
C
```

```
C INPUT
```

```
C Z1=TARGET ION CHARGE +1
```

```
C N1=INITIAL PRINCIPAL QUANTUM NUMBER
```

```
C N2=FINAL PRINCIPAL QUANTUM NUMBER
```

```
C E1=ENERGY OF EQUIVALENT ELECTRON IN RYDBERGS
```

```
C (CORRESPONDS TO ACTUAL PROJECTILE ENERGY/25KEV)
```

```
C ZP=PROJECTILE CHARGE
```

```
C ATMSSP= PROJECTILE MASS IN PROTON UNITS
```

```
C OUTPUT
```

```
C QLPR=CROSS-SECTION IN  $\pi \cdot A_0^{*2}$  UNITS
```

```
C
```

```
C
```

```
C ***** H.P.SUMMERS, JET 16/ 7/90 *****
```

```
C
```

```
C NOTES: THIS ROUTINE IS NOT YET PROPERLY ANNOTATED
```

```
C
```

```
C UNIX-IDL PORT:
```

```
C
```

```
C VERSION: 1.1 DATE: 16-1-96
```

```
C MODIFIED: TIM HAMMOND (TESSELLA SUPPORT SERVICES PLC)
```

```
C - FIRST VERSION
```

```
C
```

```
C VERSION: 1.2 DATE: 16-05-07
```

```
C MODIFIED: Allan Whiteford
```

```
C - Updated comments as part of subroutine documentation  
C procedure.
```

```
C-----
```

```
C
```

```
C-----
```

```
ZZ1=Z1*Z1
```

```
ZZP=ZP*ZP
```

```
IF (N1.LT.N2) THEN
```

```
  E=E1
```

```

        T1=1.0D0
        EN1=N1
        EN2=N2
ELSEIF (N1.EQ.N2) THEN
        QLPR=0.0D0
        RETURN
ELSE
        E2=E1-ZZ1*(1.0D0/(EN1*EN1)-1.0D0/(EN2*EN2))/(1836.12*ATMSSP)
        E=E2
        T1=(EN2*EN2*E2)/(EN1*EN1*E1)
        EN1=N2
        EN2=N1
ENDIF
S=EN2-EN1
EN12=EN1*EN2
A=2.666667D0/S*(EN2/(S*EN1))**3*(0.184D0-0.04/S**0.66667D0)*
& (1.0D0-0.2D0*S/EN12)**(1.0D0+2.0D0*S)
D=DEXP(-ZZ1*ZP/(EN12*E*E))
F=(1.0D0-0.3D0*S*D/EN12)**(1.0D0+2.0D0*S)
Y=1.0D0/(1.0D0-D*DLOG(18.0D0*S)/(4.0D0*S))
XL=DLOG((1.0D0+0.53D0*E*E*EN1*(EN2-2/EN2)/(ZZ1*ZP))
& / (1.0D0+0.4D0*E/ZP))
G=0.5D0*(E*EN1*EN1/(Z1*ZP*(EN2-1.0D0/EN2)))**3
T=DSQRT(2.0D0-(EN1/EN2)**2)
XP=2.0D0*Z1*ZP/(E*EN1*EN1*(T+1.0D0))
XM=2.0D0*Z1*ZP/(E*EN1*EN1*(T-1.0D0))
CP=(XP*XP/(2.0D0*Y+1.5D0*XP))*DLOG(1.0D0+0.66667D0*XP)
CM=(XM*XM/(2.0D0*Y+1.5D0*XM))*DLOG(1.0D0+0.66667D0*XM)
H=CM-CP
C WRITE(6,1000)E,EN1,EN2,Z1,ZP,T1
C WRITE(6,1001)A,D,XL,F,G,H
C WRITE(7,1000)E,EN1,EN2,Z1,ZP,T1
C WRITE(7,1001)A,D,XL,F,G,H
        QLPR=T1*EN1**4*(A*D*XL+F*G*H)*(ZZP/ZZ1)/E
        RETURN
1000 FORMAT(1H , 'E=' , 1PD10.2, 3X, 0P, 'EN1=' , F4.1, 3X, 'EN2=' , F4.1, 3X,
& ' Z1=' , F4.1, 3X, ' ZP=' , F4.1, 3X, ' T1=' , 1PD10.2)
1001 FORMAT(1H , 'A=' , 1PD10.2, 3X, ' D=' , 1PD10.2, 3X, ' XL=' , 1PD10.2, 3X,
& ' F=' , 1PD10.2, 3X, ' GP=' , 1PD10.2, 3X, ' H=' , 1PD10.2)
        END
INTEGER          N1,          N2
REAL*8           ATMSSP,     E1,          Z1,          ZP

```