

version 104

- [Version 1](#)

Version 1

A faire avant de la passer à Clément :

Correction rubrique BC_TR_C_PREV_NC : Utilise BC_TR_C_PREV_C

Avant

```
F_TRANCHECPREV([ BC_BRUT_PREV_C. MONTANT];[ BC_TR_A_PREV_C. MONTANT] ;[ BC_TR_B_PREV_C. MONTANT]  
;[ BC_TR_C_PREV_C. MONTANT];[ PLAFONDSECU. PLAFOND_OC];[ V_PLAFPREV. NB_TRC_NC])
```

Après

```
F_TRANCHECPREV([ BC_BRUT_PREV_NC. MONTANT];[ BC_TR_A_PREV_NC. MONTANT]  
;[ BC_TR_B_PREV_NC. MONTANT]  
;[ BC_TR_C_PREV_NC. MONTANT];[ PLAFONDSECU. PLAFOND_OC];[ V_PLAFPREV. NB_TRC_NC])
```

Mise à jour formule F_RGDU pour nouveau paramètre RGDU AD

Il s'agit d'ajouter le nouveau paramètre Heure Non exonéré.

Le paramètre Heure contrat qui était utilisé précédemment redevient Heure contrat

Avant

```
declarer (FORFAITJOUR)  
affecter (FORFAITJOUR; si (CONSTANTE( CONTRAT. SITUATIONHORAIRE) =4)  
alors (1)  
sinon (0)  
  
)  
declarer (AD)  
affecter (AD; si (CONSTANTE( CONTRAT. TYPESALARIE) =5)  
alors (1)  
sinon (0)  
  
)  
declarer (NBHEURES)  
affecter (NBHEURES; si (: AD=1)  
alors ([ AD_HNEXO. BASE])  
sinon (
```

```

        si (: FORFAITJOUR=1)
            alors (0)
        sinon (si
( CONSTANCE( CONTRAT. TYPESALARIE)=3)
            alors
( CONSTANCE( CONTRAT. HORAIRECOLLECTIF))
            sinon
( CONSTANCE( CONTRAT. HORAIREMENSUEL))
            )
        )
    )
    déclarer (NBJOURS)
    affecter (NBJOURS; si ((: FORFAITJOUR=0) ou (: AD=1))
        [] alors (0)
        [] sinon ( CONSTANCE( CONTRAT. JOUR_FORFAIT))
        )
    )
    déclarer (HSUP)
    affecter (HSUP; si (: AD=1)
        [] alors ([ AGCP_HREMUPLUS. MONTANT]*[ AD_HNEXO. BASE]/([ AD_HNEXO. BASE]+[ AD_HEX0. BASE]))
        [] sinon ([ AGCP_HREMUPLUS. MONTANT])
        )
    )
    déclarer (HORBASE)
    affecter (HORBASE; si ([ AGCPHORAN. BASE]=0)
        [] alors (
            [] si (LISTEGEN( TPSAN; CONSTANCE( CONTRAT. CATPERSO))=0)
            [] alors (1607)
            [] sinon (LISTEGEN( TPSAN; CONSTANCE( CONTRAT. CATPERSO)))
            [] )
        [] sinon ([ AGCPHORAN. BASE])
        )
    )
    déclarer (HEX0)
    affecter (HEX0; si (: AD=1)
        [] alors ([ AD_HEX0. BASE])
        [] sinon (0)
        )
    )
    déclarer (PPV)
    affecter (PPV; [ P_PPVEXO. MONTANT]+[ P_PPVNONEXO. MONTANT])
    déclarer (ANNEE)

```

```

affecter ( ANNEE; si ([ PLAFONDSECU. NUMCAS] =2)
            alors ( CONSTANTE( CONTRAT. ANNEE_CLOTURE) )
            sinon ( CONSTANTE( GENERAL. ANNEE_PAYE) ) )

declarer ( MOIS)
affecter ( MOIS; si ([ PLAFONDSECU. NUMCAS] =2)
            alors ( CONSTANTE( CONTRAT. MOIS_CLOTURE) )
            sinon ( CONSTANTE( GENERAL. MOIS_PAYE) ) )

rgdu( TYPEMONTANT;
      : ANNEE;
      : MOIS;
      CONSTANTE( GENERAL. FNAL010) ;
      : NBHEURES;
      : NBJOURS;
      : HSUP;
      [ AGCPITREMU. MONTANT] ;
      [ AGCP_NUMHREMU. MONTANT] ;
      [ B_PXHEURE. MONTANT] ;
      : HORBASE;
      : AD;
      : HEX0;
      : PPV)

```

Après

```

declarer ( FORFAITJOUR)
affecter ( FORFAITJOUR; si ( CONSTANTE( CONTRAT. SITUATIONHORAIRE) =4)
            alors ( 1)
            sinon ( 0)

)

declarer ( AD)
affecter ( AD; si ( CONSTANTE( CONTRAT. TYPESALARIE) =5)
            alors ( 1)
            sinon ( 0)

)

declarer ( NBHEURES)
affecter ( NBHEURES;
            si (: FORFAITJOUR=1)
            alors ( 0)
            sinon ( si

```

```

( CONSTANCE( CONTRAT. TYPESALARIE) =3)
                                alors
( CONSTANCE( CONTRAT. HORAIRECOLLECTIF) )
                                sinon
( CONSTANCE( CONTRAT. HORAIREMENSUEL) )
                                )
                                )
declarer ( NBJOURS)
affecter ( NBJOURS; si (( : FORFAITJOUR=0) ou ( : AD=1))
[] alors ( 0)
[] sinon ( CONSTANCE( CONTRAT. JOUR_FORFAIT) )
                                )

declarer ( HSUP)
affecter ( HSUP; si ( : AD=1)

alors([ AGCP_HREMUPLUS. MONTANT] *[ AD_HNEXO. BASE] /([ AD_HNEXO. BASE] +[ AD_HEX0. BASE] ) )
                                sinon ([ AGCP_HREMUPLUS. MONTANT] )
[] )
declarer ( HORBASE)
affecter ( HORBASE; si ([ AGCPHORAN. BASE] =0)
                                alors (
[] si ( LISTEGEN( TPSAN; CONSTANCE( CONTRAT. CATPERSO) ) =0)
[] alors ( 1607)
[] sinon ( LISTEGEN( TPSAN; CONSTANCE( CONTRAT. CATPERSO) ) )
[] )
[] sinon ([ AGCPHORAN. BASE] )
[] )
declarer ( HEX0)
affecter ( HEX0; si ( : AD=1)
                                alors ([ AD_HEX0. BASE] )
                                sinon ( 0)
                                )
declarer ( HNEXO)
affecter ( HNEXO; si ( : AD=1)
                                alors ([ AD_HNEXO. BASE] )
                                sinon ( 0)
                                )
declarer ( PPV)
affecter ( PPV; [ P_PPVEXO. MONTANT] +[ P_PPVNONE0. MONTANT] )

```

```
declarer ( ANNEE)
affecter ( ANNEE; si ([ PLAFONDSECU. NUMCAS] =2)
            alors ( CONSTANCE( CONTRAT. ANNEE_CLOTURE) )
            sinon ( CONSTANCE( GENERAL. ANNEE_PAYE) ) )
declarer ( MOIS)
affecter ( MOIS; si ([ PLAFONDSECU. NUMCAS] =2)
            alors ( CONSTANCE( CONTRAT. MOIS_CLOTURE) )
            sinon ( CONSTANCE( GENERAL. MOIS_PAYE) ) )
```

```
rgdu( TPEMONTANT;
      : ANNEE;
      : MOIS;
      CONSTANCE( GENERAL. FNAL010) ;
      : NBHEURES;
      : NBJOURS;
      : HSUP;
      [ AGCPITREMU. MONTANT] ;
      [ AGCP_NUMHREMU. MONTANT] ;
      [ B_PXHEURE. MONTANT] ;
      : HORBASE;
      : AD;
      : HEX0;
      : HNE X0;
      : PPV)
```