

version 92

- Version 1 : Apprenti
- Version 2 : Ajout rubrique CP_DIF_TRIM
- Version 3 : RGDU :
- Version 3 : RGDU : Allocation Familiale
- Version 3 RGDU : Complément maladie
- Version 3 : Problèmes sur plafond antérieur
- Version 3 : Calcul différentiel smic coefficient internat externat
- Version 4
- Version 5
- Version 6
- version 7
- Version 8
- Version 9 : Complément maladie
- Version 10 Allocation familiale
- Version 11 Code 146
- Version 12 Alloc fam Aide à domicile
- Version 13
- Version 14
- Version 15 Correctif Fonction F66AF

Version 1 : Apprenti

Lors d'un passage régime général vers apprenti après 03/2025 pour un salarié embauché avant, les nouvelles règles ne s'appliquent pas.

Il faut créer une rubrique de paye en éléments constants (APPRENTI0325) pour forcer l'application de ces règles.

Ajouter la rubrique apprenti0325 pour forcer l'embauche à mars 2025

Mettre 1 pour forcer le fait que c'est un apprenti après 2025

Ajouter une formule à la rubrique APPRENTI :

APPRENTI.APPRENTI0325

```
Si (( [ APPRENTI0325. BASE] =1)
ou (( CONSTANTE( CONTRAT. ANNEE_EMBAUCHE) *100+CONSTANTE( CONTRAT. MOIS_EMBAUCHE) >=202503) )
)
alors (1)
sinon (0)
```

V_CSG.CUM_AV_ABT

Avant

```

si (( CONSTANTE( CONTRAT. DSNINTITULE) =64) ou ( CONSTANTE( CONTRAT. DSNINTITULE) =65))
alors (
    si ( ( CONSTANTE( CONTRAT. ANNEE_EмбаUCHE) *100+CONSTANTE( CONTRAT. MOIS_EмбаUCHE) >=202503)
alors ([ B_COT_CSG_ABT. MONTANT])
sinon (0)
)
sinon (
    minimum(
        ( F_HISTOCLOT([ B_COT_CSG_ABT. MONTANT]) +[ B_COT_CSG_ABT. MONTANT])
        ;
        ( F_HISTOCLOT([ PL_MENSA. MONTANT]) +[ PL_MENSA. MONTANT]) *4
    )
)
)

```

Après

```

si (( CONSTANTE( CONTRAT. DSNINTITULE) =64) ou ( CONSTANTE( CONTRAT. DSNINTITULE) =65))
alors (
    si ([ APPRENTI. APPRENTI0325] =1)
alors ([ B_COT_CSG_ABT. MONTANT])
sinon (0)
)
sinon (
    minimum(
        ( F_HISTOCLOT([ B_COT_CSG_ABT. MONTANT]) +[ B_COT_CSG_ABT. MONTANT])
        ;
        ( F_HISTOCLOT([ PL_MENSA. MONTANT]) +[ PL_MENSA. MONTANT]) *4
    )
)
)

```

V_CSG.CUM_N_ABT

Avant

```

si (( CONSTANTE( CONTRAT. DSNINTITULE) =64) ou ( CONSTANTE( CONTRAT. DSNINTITULE) =65))
alors (
    si (
        ( CONSTANTE( CONTRAT. ANNEE_EмбаUCHE) *100+CONSTANTE( CONTRAT. MOIS_EмбаUCHE) >=202503)

```

```

        et ([ BC_BRUT_APP. MONTANT] >0)
    )
alors (
    [ B_COT_CSG_NABT. MONTANT]
)
sinon (0)
)
sinon (
    ( F_HISTOCLOT([ B_COT_CSG_ABT. MONTANT] )+[ B_COT_CSG_ABT. MONTANT] -
[V_CSG. CUM_AV_ABT] )
    +
    ( F_HISTOCLOT([ B_COT_CSG_NABT. MONTANT] )+[ B_COT_CSG_NABT. MONTANT] )
)

```

Après

```

si (( CONSTANTE( CONTRAT. DSNINTITULE) =64) ou ( CONSTANTE( CONTRAT. DSNINTITULE) =65) )
alors (
    si (
        □ ([ APPRENTI. APPRENTI0325] =1)
        et ([ BC_BRUT_APP. MONTANT] >0)
    )
alors (
    [ B_COT_CSG_NABT. MONTANT]
)
sinon (0)
)
sinon (
    ( F_HISTOCLOT([ B_COT_CSG_ABT. MONTANT] )+[ B_COT_CSG_ABT. MONTANT] -
[V_CSG. CUM_AV_ABT] )
    +
    ( F_HISTOCLOT([ B_COT_CSG_NABT. MONTANT] )+[ B_COT_CSG_NABT. MONTANT] )
)

```

CUM_ABT

Avant

```

si (( CONSTANTE( CONTRAT. DSNINTITULE) =64) ou ( CONSTANTE( CONTRAT. DSNINTITULE) =65) )
alors (
    si (

```

```

(CONSTANTE( CONTRAT. ANNEE_EмбаUCHE) *100+CONSTANTE( CONTRAT. MOIS_EмбаUCHE) >=202503)
)
alors ([ V_CSG. CUM_AV_ABT] *0. 9825)
sinon (0)
)
sinon ([ V_CSG. CUM_AV_ABT] *0. 9825)

```

Après

```

si (( CONSTANTE( CONTRAT. DSNINTITULE) =64) ou ( CONSTANTE( CONTRAT. DSNINTITULE) =65))
alors (
  si ([ APPRENTI. APPRENTI0325] =1)
  alors ([ V_CSG. CUM_AV_ABT] *0. 9825)
  sinon (0)
)
sinon ([ V_CSG. CUM_AV_ABT] *0. 9825)

```

BC_TAXESAL

Avant

```

declarer (dotaxe)
affecter (dotaxe;
  si (( CONSTANTE( CONTRAT. DSNINTITULE) =64) ou
(CONSTANTE( CONTRAT. DSNINTITULE) =65))
  alors (
    si
(( CONSTANTE( CONTRAT. ANNEE_EмбаUCHE) *100+CONSTANTE( CONTRAT. MOIS_EмбаUCHE) >=202503)
  et ([ BC_BRUT_APP. MONTANT] >0))
  alors (1)
  sinon (0)
)
  sinon (1+0)
)

si (:dotaxe=0) ou (CONSTANTE( GENERAL. EXOTAXE) =1))
alors (0)
sinon (
  si ([ BISTAXESAL. BASE] =1)
  alors ([ BC_TAXESALBIS. MONTANT])
  sinon (

```



```

alors ([ F0. TOTAL])
sinon ([ B_COT_TAXESAL. MONTANT])
)
sinon (
SI ([ EX0TAXESAL. MONTANT]=0)
ALORS ([ B_COT_TAXESAL. MONTANT])
SINON ( ( 100-[ EX0TAXESAL. MONTANT]) /100 * [ B_COT_TAXESAL. MONTANT] )
)
)
)

```

CONAIDE.SEUILAPPRENTI

Avant

```

CONSTANTE( CONTRAT. CIVJOURSTRAVAIL ) /CONSTANTE( GENERAL. NBJMOISPAYE)
*
si ( CONSTANTE( CONTRAT. ANNEE_EMBAUCHE) *100+CONSTANTE( CONTRAT. MOIS_EMBAUCHE) >=202503)
alors ( 0. 50*CONSTANTE( GENERAL. SMICHOR) *CONSTANTE( CONTRAT. HORAIRECOLLECTIF) )
sinon ( 0. 79*CONSTANTE( GENERAL. SMICMENS) )

```

Après

```

CONSTANTE( CONTRAT. CIVJOURSTRAVAIL ) /CONSTANTE( GENERAL. NBJMOISPAYE)
*
si ([ APPRENTI. APPRENTI0325]=1)
alors ( 0. 50*CONSTANTE( GENERAL. SMICHOR) *CONSTANTE( CONTRAT. HORAIRECOLLECTIF) )
sinon ( 0. 79*CONSTANTE( GENERAL. SMICMENS) )

```

Version 2 : Ajout rubrique CP_DIF_TRIM

c'est une copie de CP_DIF.

Le libellé est Indemnité différentielle de congés trimestriels

Version 3 : RGDU :

Le calcul de la RGDU

Il s'agit de remplacer les formules base de cotisation de l'agcp par le mot clé RGDU

Pour simplifier, ajout d'une fonction avec un paramètre : TPEMONTANT

- 0-> RGDU mensuel
- 1-> RGDU Urssaf mensuel
- 2-> RGDU AA mensuel

F_RGDU

```
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 (CONSTANTE( CONTRAT. HORAIREMENSUEL))
        )
)
)
declarer (NBjours)
affecter (NBjours; si ((: FORFAITJOUR=0) ou (: AD=1))
        alors (0)
        sinon (CONSTANTE( 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; CONSTANTE( CONTRAT. CATPERSO) ) =0)
[[[alors (1607)
[[[sinon (LISTEGEN( TPSAN; CONSTANTE( CONTRAT. CATPERSO) ))
[[[)
[[[ sinon ([ AGCPHORAN. BASE] )
) )
declarer (HEX0)
affecter (HEX0; si (: AD=1)
        alors ([ AD_HEX0. BASE] )
        sinon (0)
)
)
declarer (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( TPEMONTANT;
      : ANNEE;
      : MOIS;
      CONSTANTE( GENERAL. FNAL010) ;
      : NBHEURES;
      : NBJOURS;
      : HSUP;
      [ AGCPI TREMU. MONTANT] ;
      [ AGCP_NUMHREMU. MONTANT] ;
      [ B_PXHEURE. MONTANT] ;

```

```
: HORBASE;  
: AD;  
: HEX0)
```

BC_AGCPURSSAF

Avant

```
si (  
  ([ AGCPURSSAF_1. BASE]=0) et ([ AGCPURSAFREG_1. BASE]=0) et ([ AGCPRUAA_1. BASE]=0)  
  et ([ AGCPURS_AD_1. BASE]=0) et ([ AGCPURFR_AD_1. BASE]=0) et ([ AGCPRUAA_AD_1. BASE]=0)  
  )  
alors (  
  si (( CONSTANTE( CONTRAT. ANNEE_CLOTURE)=0)  
      ou ( CONSTANTE( CONTRAT. ANNEE_CLOTURE)>=2024) )  
  alors ([ V_NEWAGCP. AGCPURSSAF])  
  sinon (  
    [ V_AGCP. TOTAGCP]  
    *CONSTANTE( GENERAL. COEFAGCPURSSAF)  
    /( CONSTANTE( GENERAL. COEFAGCPURSSAF)+CONSTANTE( GENERAL. COEFAGRETRAITE) )  
    +[ V_AGCP. TOTAGCPCHOM]  
  
    histocumulsitu([ BC_AGCPURSSAF. MONTANT];1; CONSTANTE( CONTRAT. ANNEE_CLOTURE);12; CONSTANTE( CONTRAT.  
      )  
    )  
  )  
sinon (0)
```

Après

```
F_RGDU(1)
```

BC_AGCPAA

Avant

```
si (  
  ([ AGCPURSSAF_1. BASE]=0) et ([ AGCPURSAFREG_1. BASE]=0) et ([ AGCPRUAA_1. BASE]=0)  
  et ([ AGCPURS_AD_1. BASE]=0) et ([ AGCPURFR_AD_1. BASE]=0) et ([ AGCPRUAA_AD_1. BASE]=0)  
  )  
alors (  
  )
```

```

si (( CONSTANTE( CONTRAT. ANNEE_CLOTURE) =0)
    ou ( CONSTANTE( CONTRAT. ANNEE_CLOTURE) >=2024) )
alors ([ V_NEWAGCP. AGCPAA] )
sinon (
    si ([ AGCPURSSAF. BASE] <>0)
    alors (
        [ V_AGCP. MTAGCPMENS] +[ V_AGCP. MTAGCPMENSCHOM] -
[ AGCPURSSAF. BASE]
    )
    sinon (0)
    +
    si ([ AGCPURSSRENF. BASE] <>0)
    alors (
        [ V_AGCP. MTAGCPMENSRENF] - [ AGCPURSSRENF. BASE]
    )
    sinon (0)
    )
)
sinon (0)

```

Après

```
F_RGDU(2)
```

BC_ADAGCPURSS

Avant

```

si ( CONSTANTE( CONTRAT. RAPPEL_FICHE) =1)
alors (0)
sinon (
    si ((([ AGCPURSSAF_1. BASE]=0) et ([ AGCPURSAFREG_1. BASE]=0) et
([ AGCPRUAA_1. BASE]=0)
        et ([ AGCPURS_AD_1. BASE]=0) et ([ AGCPURFR_AD_1. BASE]=0) et
([ AGCPRUAA_AD_1. BASE]=0))
    alors (
        si (( CONSTANTE( CONTRAT. ANNEE_CLOTURE) =0)
            ou ( CONSTANTE( CONTRAT. ANNEE_CLOTURE) >=2024) )
        alors ([ V_AD_AGCP. ADAGCPURSSAF] )
        sinon (0)
    )
)

```

```
sinon (0)
)
```

Après

```
F_RGDU(1)
```

BC_ADAGCPRUAA

Avant

```
si ((
  ([ AGCPURSSAF_1. BASE]=0) et ([ AGCPURSAFREG_1. BASE]=0) et ([ AGCPRUAA_1. BASE]=0)
  et ([ AGCPURS_AD_1. BASE]=0) et ([ AGCPURFR_AD_1. BASE]=0) et ([ AGCPRUAA_AD_1. BASE]=0)
)
ou ( CONSTANTE( CONTRAT. RAPPEL_FICHE)=0)
)
alors (
  si (( CONSTANTE( CONTRAT. ANNEE_CLOTURE)=0)
    ou ( CONSTANTE( CONTRAT. ANNEE_CLOTURE)>=2024) )
  alors ([ V_AD_AGCP. ADAGCPAA])
  sinon (0)
)
sinon (0)
```

Après

```
F_RGDU(2)
```

Version 3 : RGDU :

Allocation Familiale

Le calcul de l'allocation familiale

Plus de régularisation en 2026.

Plus d'alloc taux réduit non plus

BC_REG_ALLOC

Avant

```
declarer (TXSMIC)
affecter (TXSMIC;
    si (([PLAFONDSECU. NUMCAS]=2) et (CONSTANTE(CONTRAT. ANNEE_CLOTURE) <2025))
    alors (3.5)
    sinon (3.3)
    )
si (
    ([V_NEWAGCP. CUMREMUAF] <=( : TXSMIC*[V_NEWAGCP. CUMSMICPR03112] ))
    ou ([V_NEWAGCP. CUMREMUAF] <=0)
    ou ([V_NEWAGCP. CUMSMICPR03112] <=0)
)
alors (
    histocumulsitu([V_ALLOC. BASEALLOC])*(-1)
    -histocumulsitu([BC_REG_ALLOC. MONTANT])
)
sinon (
    histocumulsitu([V_ALLOC. BASEALLOCRED])
    +histocumulsitu([BC_REG_ALLOCRE. MONTANT])
)
```

Après

```
si (([PLAFONDSECU. NUMCAS]=2) et (CONSTANTE(CONTRAT. ANNEE_CLOTURE) <2026))
```

```

alors (
  declarer ( TXSMIC)
  affecter ( TXSMIC;
            si (([ PLAFONDSECU. NUMCAS]=2) et
(CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2025) )
            alors ( 3.5)
            sinon ( 3.3)
            )
  si (
    ([ V_NEWAGCP. CUMREMUAF] <=( : TXSMIC*[ V_NEWAGCP. CUMSMICPR03112] ))
    ou ([ V_NEWAGCP. CUMREMUAF] <=0)
    ou ([ V_NEWAGCP. CUMSMICPR03112] <=0)
    )
  alors (
    histocumulsitu([ V_ALLOC. BASEALLOC] )*(-1)
    -histocumulsitu([ BC_REG_ALLOC. MONTANT] )
    )
  sinon (
    histocumulsitu([ V_ALLOC. BASEALLOCRED] )
    +histocumulsitu([ BC_REG_ALLOCRE. MONTANT] )
    )
  )
sinon ( 0
)
)

```

ALLOCFAMREDUIT.TXEMP

Avant

```

si (
  ([ V_NEWAGCP. CUMREMUAF] >[ V_NEWAGCP. TXSMICAF] *[ V_NEWAGCP. CUMSMICPR03112] )
  et ([ V_NEWAGCP. CUMREMUAF] >0)
  et ([ V_NEWAGCP. CUMSMICPR03112] >0)
  )
alors ( 0)
sinon ( LISTEGEN( TX_ALLOCF; 0002) )
)

```

Après

```

si (([ PLAFONDSECU. NUMCAS]=2) et ( CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2026) )
alors (

```

```

si (
  ([ V_NEWAGCP. CUMREMUAF ] > [ V_NEWAGCP. TXSMICAF ] * [ V_NEWAGCP. CUMSMICPRO3112 ] )
  et ([ V_NEWAGCP. CUMREMUAF ] > 0)
  et ([ V_NEWAGCP. CUMSMICPRO3112 ] > 0)
)
alors ( 0)
sinon ( LISTEGEN( TX_ALLOCF; 0002) )
)
sinon ( 0)

```

ALLOCFAM.TXEMP

Avant

```

si (
  ([ V_NEWAGCP. CUMREMUAF ] > [ V_NEWAGCP. TXSMICAF ] * [ V_NEWAGCP. CUMSMICPRO3112 ] )
  et ([ V_NEWAGCP. CUMREMUAF ] > 0)
  et ([ V_NEWAGCP. CUMSMICPRO3112 ] > 0)
)
alors ( LISTEGEN( TX_ALLOCF; 0001) )
sinon ( 0)

```

Après

```

si (( [ PLAFONDSECU. NUMCAS ] = 2) et ( CONSTANTE( CONTRAT. ANNEE_CLOTURE ) < 2026) )
alors (
  si (
    ([ V_NEWAGCP. CUMREMUAF ] > [ V_NEWAGCP. TXSMICAF ] * [ V_NEWAGCP. CUMSMICPRO3112 ] )
    et ([ V_NEWAGCP. CUMREMUAF ] > 0)
    et ([ V_NEWAGCP. CUMSMICPRO3112 ] > 0)
  )
  alors ( LISTEGEN( TX_ALLOCF; 0001) )
  sinon ( 0)
)
sinon ( 0)

```

ALLOCFAMCOMP.TXEMP

Avant

```

si (

```

```

([ V_NEWAGCP. CUMREMUAF ] > [ V_NEWAGCP. TXSMICAF ] * [ V_NEWAGCP. CUMSMICPR03112 ] )
et ([ V_NEWAGCP. CUMREMUAF ] > 0)
et ([ V_NEWAGCP. CUMSMICPR03112 ] > 0)
)
alors ( 0) sinon ( 1. 80)

```

Après

```

si (( [ PLAFONDSECU. NUMCAS ] = 2) et ( CONSTANTE( CONTRAT. ANNEE_CLOTURE) < 2026))
alors (
  si (
    ([ V_NEWAGCP. CUMREMUAF ] > [ V_NEWAGCP. TXSMICAF ] * [ V_NEWAGCP. CUMSMICPR03112 ] )
    et ([ V_NEWAGCP. CUMREMUAF ] > 0)
    et ([ V_NEWAGCP. CUMSMICPR03112 ] > 0)
  )
  alors ( 0)
  sinon ( 1. 80)
)
sinon ( 0)

```

ALLOCFAMCOMPNEG.TXEMP

Avant

```

si (
  ([ V_NEWAGCP. CUMREMUAF ] > [ V_NEWAGCP. TXSMICAF ] * [ V_NEWAGCP. CUMSMICPR03112 ] )
  et ([ V_NEWAGCP. CUMREMUAF ] > 0)
  et ([ V_NEWAGCP. CUMSMICPR03112 ] > 0)
)
alors ( 0) sinon ( -1. 80)

```

Après

```

si (( [ PLAFONDSECU. NUMCAS ] = 2) et ( CONSTANTE( CONTRAT. ANNEE_CLOTURE) < 2026))
alors (
  si (
    ([ V_NEWAGCP. CUMREMUAF ] > [ V_NEWAGCP. TXSMICAF ] * [ V_NEWAGCP. CUMSMICPR03112 ] )
    et ([ V_NEWAGCP. CUMREMUAF ] > 0)
    et ([ V_NEWAGCP. CUMSMICPR03112 ] > 0)
  )
  alors ( 0)

```

```
sinon (-1.80)  
)  
sinon (0)
```

V_ALLOC

Toujours calculé à False

Mise à zéro de toutes les formules sauf BASEALLOC

Version 3 RGDU :

Complément maladie

Il y a toujours un complément maladie

BC_MALCOMP25

Avant

```
si (
  (([ V_NEWAGCP. CUMREMUAF]) > ([ V_NEWAGCP. TXSMICMAL] * [ V_NEWAGCP. CUMSMICPR03112]))
  et ([ V_NEWAGCP. CUMREMUAF] > 0)
  et ([ V_NEWAGCP. CUMSMICPR03112] > 0)
)
alors (
  f_histoclotsit([ BC_BRUT_URSAFF. MONTANT]) + [ BC_BRUT_URSAFF. MONTANT]
  - f_histoclotsit([ BC_MALCOMP25. MONTANT])
)
sinon (
  f_histoclotsit([ BC_MALCOMP25. MONTANT]) * (-1)
)
```

Après

```
si (([ PLAFONDSECU. NUMCAS] = 2) et (CONSTANTE( CONTRAT. ANNEE_CLOTURE) < 2026))
alors (
  si (
    (([ V_NEWAGCP. CUMREMUAF]) >
    ([ V_NEWAGCP. TXSMICMAL] * [ V_NEWAGCP. CUMSMICPR03112]))
    et ([ V_NEWAGCP. CUMREMUAF] > 0)
    et ([ V_NEWAGCP. CUMSMICPR03112] > 0)
  )
  alors (
    f_histoclotsit([ BC_BRUT_URSAFF. MONTANT]) + [ BC_BRUT_URSAFF. MONTANT]
    - f_histoclotsit([ BC_MALCOMP25. MONTANT])
  )
)
```

```
sinon (
    f_histoclotsit([ BC_MALCOMP25. MONTANT])*(-1)
)
)
sinon ([ BC_BRUT_URSAFF. MONTANT])
```

MALCOMP25.TXEMP

Avant

```
si ([ MALCOMP25. BASE]>0)
alors (6)
sinon ( si ([ BC_BRUT_STA. MONTANT]>0) et ( CONSTANTE( CONTRAT. DSNATURE)=29) )
    alors (6)
    sinon (0)
)
```

Après

```
si ([ PLAFONDSECU. NUMCAS]=2) et ( CONSTANTE( CONTRAT. ANNEE_CLOTURE)<2026) )
alors (
    si ([ MALCOMP25. BASE]>0)
    alors (6)
    sinon ( si ([ BC_BRUT_STA. MONTANT]>0) et
( CONSTANTE( CONTRAT. DSNATURE)=29) )
        alors (6)
        sinon (0)
    )
)
sinon (6)
)
```

Version 3 : Problèmes sur plafond antérieur

Alerte de Christophe sur PLAFONDS_A : Utilise une constante qui n'existe plus :

PLAFONDS_A : Je le met à toujours calculé non

PLAFONDS_A.PLA_MOIS

Avant

```
SI (( CONSTANTE( General. PRORATAPLAFOND) =1)
    ou ( CONSTANTE( GENERAL. ANNEE_PAYE) *100+CONSTANTE( GENERAL. MOIS_PAYE) >=201807)
    )
alors ( maximum( 0; MINIMUM( CONSTANTE( GENERAL. PLFSECUMOIS) ; [ PLAFONDS_A. NEW_PLA_MOIS] ) ) )
sinon( maximum( 0; MINIMUM( CONSTANTE( GENERAL. PLFSECUMOIS) ; [ PLAFONDS_A. ANC_PLA_MOIS] ) ) ) )
```

Après

0

PLAFONDS_A.PLA_MOISANT

Avant

```
SI (( CONSTANTE( General. PRORATAPLAFOND) =1)
    ou ( CONSTANTE( GENERAL. ANNEE_PAYE) *100+CONSTANTE( GENERAL. MOIS_PAYE) >=201807)
    )
alors ( maximum( 0; MINIMUM( CONSTANTE( GENERAL. PLFSECUMOIS) ; [ PLAFONDS_A. NEW_PLA_MOISANT] ) ) )
sinon( maximum( 0; MINIMUM( CONSTANTE( GENERAL. PLFSECUMOIS) ; [ PLAFONDS_A. ANC_PLA_MOISANT] ) ) ) )
```

Après

0

ANC_PLA_MOISANT

Avant

```
si (( CONSTANTE( GENERAL. JRS_ANT)=1) et ( CONSTANTE( CONTRAT. NEUTRAL_PLAF) >0))
alors (
  si ( Constante(General. Mois_paye)=2)
  alors ( si ([ JRSPLAFOND. MONTANT] >=28)
    alors ([ PLAFONDS_A. PLA_MOIS] / 30)
    sinon ([ PLAFONDS_A. PLA_MOIS] / [ JRSPLAFOND. MONTANT])
  )
  sinon ( [ PLAFONDS_A. PLA_MOIS] / minimum([ JRSPLAFOND. MONTANT]; 30))
  *
  [ ANT_JOURS. BASE]
  +
  si ([ ANT_HEURES. MONTANT] <>0)
  alors (
    minimum( CONSTANTE( GENERAL. PLFSECUMOIS);
CONSTANTE( GENERAL. PLFSECUMOIS) * [ ANT_HEURES. BASE] / CONSTANTE( CONTRAT. HORAIRECOLLECTIF) )
    )
  sinon ( 0)
  )
sinon ( 0)
```

Après

0

NEW_PLA_MOISANT

Avant

```
si ( CONSTANTE( GENERAL. JRS_ANT) =1)
alors (
  minimum( CONSTANTE( GENERAL. PLFSECUMOIS);
CONSTANTE( GENERAL. PLFSECUMOIS)
  *
  ( si ([ PL_PROTPSPAR. MONTANT] =0)
    alors (1)
    sinon (
      si ( CONSTANTE( CONTRAT. TYPEPLAFOND) =4)
      alors (
```

```

                CONSTANTE( CONTRAT. TAUXPLAFOND) /100
            )
        sinon (
            ( CONSTANTE( CONTRAT. HORAIREMENSUEL) +[ ANT_HEURES. MONTANT] )
              /
            CONSTANTE( CONTRAT. HORAIRECOLLECTIF)
              )
        )
    )
    *(
        minimum( maximum( 0; [ ANT_JOURS. BASE] ); CONSTANTE( GENERAL. NBJMOISPAYE_1) )
        /CONSTANTE( GENERAL. NBJMOISPAYE_1)
    )
)
sinon ( 0)

```

Après

0

Version 3 : Calcul différentiel smic coefficient internat externat

66_DIFINTEXT

Différence internatexternat

Avant

```
CONSTANTE( CONTRAT. HORAIREMENSUEL ) /CONSTANTE( CONTRAT. HORAIRECOLLECTIF)
*
si ([ COEFEXT66. COEFEXT66] =0)
alors (0)
sinon (
maximum(
([ COEFREF. COEFREF] - [ COEFEXT66. COEFEXT66] ) * [ VP. VP]
* ( 1+CONSTANTE( GENERAL. TAUXSUJETION66) /100)
;0
)
)
```

Après

```
CONSTANTE( CONTRAT. HORAIREMENSUEL ) /CONSTANTE( CONTRAT. HORAIRECOLLECTIF)
*
si ([ COEFEXT66. COEFEXT66] =0)
alors (0)
sinon (
declarer (coefext)
affecter(coefext; COEFANCEXT( CONSTANTE( CONTRAT. IDGRILLE) ; [ 66_ECHELON. MONTANT] ))
maximum(
si (:coefext=0)
alors ([ COEFREF. COEFREF] - [ COEFEXT66. COEFEXT66] )
```

```
sinon (  
    ([ COEFANC. COEFANC] -: coefext)  
    )  
* [ VP. VP]  
* ( 1+CONSTANTE( GENERAL. TAUXSUJETION66) /100)  
;0  
)  
)
```

Version 4

VMRR taux réduit

TRANSPVMRRMED.TXEMP

Avant

0

Après

Non modifiable

```
LISTEGEN( TX_TRANSPVMRR; CONSTANTE( CONTRAT. TAUX_TAXETRANSPORTVMRR ) )
```

Version 5

Correction pour AGCP AD

BC_ADAGCPRUAA

```
F_RGDU(12)
```

BC_ADAGCPURSS

```
F_RGDU(11)
```

On va créer deux bases de cotisation correspondant au brut urssaf AD exo et brut urssaf AD nexo

BC_BRUT_ADEXO

```
si ( CONSTANTE( CONTRAT. TYPESALARIE) =5)
alors ( [ AGCPI TREMU. MONTANT] * [ AD_HEXO. BASE] / ( [ AD_HEXO. BASE] + [ AD_HNEXO. BASE] ) )
sinon ( 0)
```

BC_BRUT_ADNEXO

```
si ( CONSTANTE( CONTRAT. TYPESALARIE) =5)
alors ( [ AGCPI TREMU. MONTANT] * [ AD_HNEXO. BASE] / ( [ AD_HEXO. BASE] + [ AD_HNEXO. BASE] ) )
sinon ( 0)
```

Complément maladie

BC_MALCOMP25

En 2026, pour les AD, on devrait avoir

- Toujours le complément maladie pour la partie non exo
- Pour la partie non exo, Complément maladie si la rému > 2.25SMIC

Mais on va considérer qu'un AD ne gagne jamais au dessus de 2.25SMIC

Il paiera juste le complément sur la partie non exo

```

declarer (PASADREMU)
affecter
(PASADREMU; [ AGCPITREMU. MONTANT] * [ AD_HNEXO. BASE] / ([ AD_HEXO. BASE] + [ AD_HNEXO. BASE] )
)
si (
(( [ PLAFONDSECU. NUMCAS] =2) et ( CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2026) )
ou ( CONSTANTE( GENERAL. ANNEE_PAYE) <2026)
)
alors (
si (
(( [ V_NEWAGCP. CUMREMUAF] ) >
[ [ V_NEWAGCP. TXSMICMAL] * [ V_NEWAGCP. CUMSMICPR03112] ) )
et ( [ V_NEWAGCP. CUMREMUAF] >0)
et ( [ V_NEWAGCP. CUMSMICPR03112] >0)
)
alors (
f_histoclotsit([ BC_BRUT_URSAFF. MONTANT] ) + [ BC_BRUT_URSAFF. MONTANT]
- f_histoclotsit([ BC_MALCOMP25. MONTANT] )
)
sinon (
f_histoclotsit([ BC_MALCOMP25. MONTANT] ) * (-1)
)
)
sinon (
si ( CONSTANTE( CONTRAT. TYPESALARIE) =5)
alors (: PASADREMU)
sinon ([ BC_BRUT_URSAFF. MONTANT] )
)
)

```

Allocation familiale

En 2026, pour les AD, on devrait avoir

- Toujours l'allocation familiale taux plein pour la partie non exo
- Pour la partie non exo, l'allocation familiale taux plein si la rému > 3.3SMIC

La encore on considère que le salarié a une rémunération <3.3 SMIC

Du coup rien à changer dans les rubriques

Il suffira d'appliquer la base BC_BRUT_ADNEXO pour le régime AD et pour la rubrique ALLOCFAM

Version 6

Correctif ALLOCFAM.TXEMP

Avant

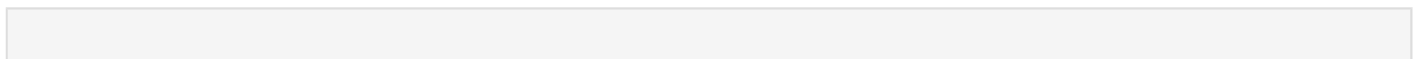
```
si (([PLAFONDSECU. NUMCAS]=2) et (CONSTANTE(CONTRAT. ANNEE_CLOTURE) <2026))
alors (
  si (
    ([ V_NEWAGCP. CUMREMUAF] > [ V_NEWAGCP. TXSMICAF] * [ V_NEWAGCP. CUMSMICPRO3112])
    et ([ V_NEWAGCP. CUMREMUAF] > 0)
    et ([ V_NEWAGCP. CUMSMICPRO3112] > 0)
  )
  alors (LISTEGEN(TX_ALLOCF; 0001))
  sinon (0)
)
sinon (LISTEGEN(TX_ALLOCF; 0001))
```

Après

```
si (
  (([PLAFONDSECU. NUMCAS]=2) et (CONSTANTE(CONTRAT. ANNEE_CLOTURE) <2026))
  ou (CONSTANTE(GENERAL. ANNEE_PAYE) <2026)
)
alors (
  si (
    ([ V_NEWAGCP. CUMREMUAF] > [ V_NEWAGCP. TXSMICAF] * [ V_NEWAGCP. CUMSMICPRO3112])
    et ([ V_NEWAGCP. CUMREMUAF] > 0)
    et ([ V_NEWAGCP. CUMSMICPRO3112] > 0)
  )
  alors (LISTEGEN(TX_ALLOCF; 0001))
  sinon (0)
)
sinon (LISTEGEN(TX_ALLOCF; 0001))
```

ALLOCFAMREDUIT

Avant



```

si (([PLAFONDSECU. NUMCAS]=2) et (CONSTANTE(CONTRAT. ANNEE_CLOTURE)<2026))
alors (
  si (
    ([ V_NEWAGCP. CUMREMUAF]>[ V_NEWAGCP. TXSMICAF]*[ V_NEWAGCP. CUMSMICPR03112])
    et ([ V_NEWAGCP. CUMREMUAF]>0)
    et ([ V_NEWAGCP. CUMSMICPR03112]>0)
  )
  alors (0)
  sinon (LISTEGEN(TX_ALLOCF;0002))
)
sinon (0)

```

Après

```

si (
  (([PLAFONDSECU. NUMCAS]=2) et (CONSTANTE(CONTRAT. ANNEE_CLOTURE)<2026))
  ou (CONSTANTE(GENERAL. ANNEE_PAYE)<2026)
)
alors (
  si (
    ([ V_NEWAGCP. CUMREMUAF]>[ V_NEWAGCP. TXSMICAF]*[ V_NEWAGCP. CUMSMICPR03112])
    et ([ V_NEWAGCP. CUMREMUAF]>0)
    et ([ V_NEWAGCP. CUMSMICPR03112]>0)
  )
  alors (0)
  sinon (LISTEGEN(TX_ALLOCF;0002))
)
sinon (0)

```

version 7

Correction pour le budget de merde

ALLOCFAM.TXEMP

Avant

```
si (
  (([PLAFONDSECU. NUMCAS]=2) et (CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2026))
  ou (CONSTANTE( GENERAL. ANNEE_PAYE) <2026)
)
alors (
  si (
    ([ V_NEWAGCP. CUMREMUAF] >[ V_NEWAGCP. TXSMICAF] *[ V_NEWAGCP. CUMSMICPR03112])
    et ([ V_NEWAGCP. CUMREMUAF] >0)
    et ([ V_NEWAGCP. CUMSMICPR03112] >0)
  )
  alors (LISTEGEN( TX_ALLOCF; 0001))
  sinon (0)
)
sinon (LISTEGEN( TX_ALLOCF; 0001))
```

Après

```
si (CONSTANTE( GENERAL. ANNEE_PAYE) >=2026)
alors (
  LISTEGEN( TX_ALLOCF; 0001)
)
sinon (
  si ((([PLAFONDSECU. NUMCAS]=2) et (CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2026))
  alors (
    si (
      (([ V_NEWAGCP. CUMREMUAF] ) >
([ V_NEWAGCP. TXSMICMAL] *[ V_NEWAGCP. CUMSMICPR03112]))
      et ([ V_NEWAGCP. CUMREMUAF] >0)
      et ([ V_NEWAGCP. CUMSMICPR03112] >0)
    )
  )
)
```

```

        alors (0)
        sinon (LISTEGEN(TX_ALLOCF;0001))
    )
sinon (LISTEGEN(TX_ALLOCF;0001))
)

```

ALLOCFAMREDUIT

Avant

```

si (
    (([PLAFONDSECU. NUMCAS]=2) et (CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2026))
    ou ( CONSTANTE( GENERAL. ANNEE_PAYE) <2026)
)
alors (
    si (
        ([ V_NEWAGCP. CUMREMUAF] >[ V_NEWAGCP. TXSMICAF] *[ V_NEWAGCP. CUMSMICPR03112])
        et ([ V_NEWAGCP. CUMREMUAF] >0)
        et ([ V_NEWAGCP. CUMSMICPR03112] >0)
    )
    alors (0)
    sinon (LISTEGEN(TX_ALLOCF;0002))
)
sinon (0)

```

Après

```

si ( CONSTANTE( GENERAL. ANNEE_PAYE) >=2026)
alors (
    0
)
sinon (
    si ((([PLAFONDSECU. NUMCAS]=2) et (CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2026))
        alors (
            si (
                (([ V_NEWAGCP. CUMREMUAF] ) >
                ([ V_NEWAGCP. TXSMICMAL] *[ V_NEWAGCP. CUMSMICPR03112]))
                et ([ V_NEWAGCP. CUMREMUAF] >0)
                et ([ V_NEWAGCP. CUMSMICPR03112] >0)
            )
            alors (0)

```

```

        sinon (LISTEGEN(TX_ALLOCF;0002))
    )
    sinon (0)
)

```

BC_MALCOMP25

Avant

```

declarer (PASADREMU)
affecter
(PASADREMU; [ AGCPITREMU. MONTANT] * [ AD_HNEXO. BASE] / ([ AD_HEXO. BASE] + [ AD_HNEXO. BASE] )
)
si (
    (( [ PLAFONDSECU. NUMCAS] =2) et ( CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2026) )
    ou ( CONSTANTE( GENERAL. ANNEE_PAYE) <2026)
)
alors (
    si (
        (( [ V_NEWAGCP. CUMREMUAF] ) >
        ([ V_NEWAGCP. TXSMICMAL] * [ V_NEWAGCP. CUMSMICPR03112] ) )
        et ( [ V_NEWAGCP. CUMREMUAF] >0)
        et ( [ V_NEWAGCP. CUMSMICPR03112] >0)
    )
    alors (
        f_histoclotsit([ BC_BRUT_URSAFF. MONTANT] ) + [ BC_BRUT_URSAFF. MONTANT]
        - f_histoclotsit([ BC_MALCOMP25. MONTANT] )
    )
    sinon (
        f_histoclotsit([ BC_MALCOMP25. MONTANT] ) * (-1)
    )
)
sinon (
    si ( CONSTANTE( CONTRAT. TYPESALARIE) =5)
    alors (: PASADREMU)
    sinon ([ BC_BRUT_URSAFF. MONTANT] )
)
)

```

Après

```

declarer (REMU)

```

```

affecter (REMU; si (CONSTANTE( CONTRAT. TYPESALARIE)=5)
          alors
          ([ AGCPITREMU. MONTANT]*[ AD_HNEXO. BASE]/([ AD_HEXO. BASE]+[ AD_HNEXO. BASE]))
          sinon ([ BC_BRUT_URSAFF. MONTANT])
        )
si (CONSTANTE( GENERAL. ANNEE_PAYE)>=2026)
alors (
      :remu
    )
sinon (
      si ([[ PLAFONDSECU. NUMCAS]=2) et (CONSTANTE( CONTRAT. ANNEE_CLOTURE)<2026))
      alors (
            si (
                  ([[ V_NEWAGCP. CUMREMUAF]) >
([ V_NEWAGCP. TXSMICMAL]*[ V_NEWAGCP. CUMSMICPR03112]))
                  et ([ V_NEWAGCP. CUMREMUAF]>0)
                  et ([ V_NEWAGCP. CUMSMICPR03112]>0)
                )
            alors (
                  f_histoclotsit([ BC_BRUT_URSAFF. MONTANT])+
[ BC_BRUT_URSAFF. MONTANT]
                  - f_histoclotsit([ BC_MALCOMP25. MONTANT])
                )
            sinon (
                  f_histoclotsit([ BC_MALCOMP25. MONTANT])*(-1)
                )
            )
      sinon (:remu)
    )

```

MALCOMP25.TXEMP

La passer en non modifiable

Dans le budget il faudra l'enlever et la remettre

Avant

```

si ([[ PLAFONDSECU. NUMCAS]=2) et (CONSTANTE( CONTRAT. ANNEE_CLOTURE)<2026))
alors (
      si ([ MALCOMP25. BASE]>0)

```

```

    alors (6)
    sinon ( si ([BC_BRUT_STA.MONTANT]>0) et
(CONSTANTE( CONTRAT. DSNNATURE) =29)
        alors (6)
        sinon (0)
    )
)
sinon (6
)

```

Après

```

si ( CONSTANTE( GENERAL. ANNEE_PAYE) >=2026)
alors (
    6
)
sinon (
    si ([ PLAFONDSECU. NUMCAS] =2) et ( CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2026) )
    alors (
        si ([ MALCOMP25. BASE] >0)
        alors (6)
        sinon ( si ([BC_BRUT_STA.MONTANT]>0) et
(CONSTANTE( CONTRAT. DSNNATURE) =29)
            alors (6)
            sinon (0)
        )
    )
    sinon (6)
)

```



```

        alors (
[[[si ( LISTEGEN( TPSAN; CONSTANTE( CONTRAT. CATPERSO ) =0)
[[[alors ( 1607)
[[[sinon ( LISTEGEN( TPSAN; CONSTANTE( CONTRAT. CATPERSO ) )
[[[
[[[ sinon ( [ AGCPHORAN. BASE] )
[[[ )
declarer ( HEX0)
affecter ( HEX0; si (: AD=1)
            alors ( [ AD_HEX0. BASE] )
            sinon ( 0)
        )
declarer ( 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( TPEMONTANT;
    : ANNEE;
    : MOIS;
    CONSTANTE( GENERAL. FNAL010) ;
    : NBHEURES;
    : NBJOURS;
    : HSUP;
    [ AGCPITREMU. MONTANT] ;
    [ AGCP_NUMHREMU. MONTANT] ;
    [ B_PXHEURE. MONTANT] ;
    : HORBASE;
    : AD;
    : HEX0)

```

Version 9 : Complément maladie

BC_MALCOMP25 on est forcément après 2026

```
si ([[PLAFONDSECU. NUMCAS]=2) et (CONSTANTE(CONTRAT. ANNEE_CLOTURE) <2026))
alors (
  si (
    ([[V_NEWAGCP. CUMREMUAF]) >
    ([[V_NEWAGCP. TXSMICMAL]*[V_NEWAGCP. CUMSMICPRO3112]])
    et ([V_NEWAGCP. CUMREMUAF] >0)
    et ([V_NEWAGCP. CUMSMICPRO3112] >0)
  )
  alors (
    f_histoclotsit([BC_BRUT_URSAFF. MONTANT]) + [BC_BRUT_URSAFF. MONTANT]
    - f_histoclotsit([BC_MALCOMP25. MONTANT])
  )
  sinon (
    f_histoclotsit([BC_MALCOMP25. MONTANT]) * (-1)
  )
)
sinon ( si (CONSTANTE(CONTRAT. TYPESALARIE) =5)
  alors ([AGCPITREMU. MONTANT] * [AD_HNEXO. BASE] / ([AD_HEX0. BASE] + [AD_HNEXO. BASE]))
  sinon ([BC_BRUT_URSAFF. MONTANT])
)
```

MALCOMP25.TXEMP

```
si ([MALCOMP25. BASE] >0)
alors (6)
sinon ( si ([[BC_BRUT_STA. MONTANT] >0) et (CONSTANTE(CONTRAT. DSNATURE) =29))
  alors (6)
  sinon (0)
)
```

MALCOMP25NREG.TXEMP : On ne change pas

Version 10 Allocation familiale

ALLOCFAM.TXEMP

```
si (([PLAFONDSECU. NUMCAS]=2) et (CONSTANTE(CONTRAT. ANNEE_CLOTURE) <2026))
alors (
  si (
    (([V_NEWAGCP. CUMREMUAF]) >
([V_NEWAGCP. TXSMICMAL]*[V_NEWAGCP. CUMSMICPR03112]))
    et ([V_NEWAGCP. CUMREMUAF]>0)
    et ([V_NEWAGCP. CUMSMICPR03112]>0)
  )
  alors (LISTEGEN(TX_ALLOCF;0001))
  sinon (0)
)
sinon (LISTEGEN(TX_ALLOCF;0001))
```

ALLOCFAMREDUIT.TXEMP

```
si (([PLAFONDSECU. NUMCAS]=2) et (CONSTANTE(CONTRAT. ANNEE_CLOTURE) <2026))
alors (
  si (
    (([V_NEWAGCP. CUMREMUAF]) >
([V_NEWAGCP. TXSMICMAL]*[V_NEWAGCP. CUMSMICPR03112]))
    et ([V_NEWAGCP. CUMREMUAF]>0)
    et ([V_NEWAGCP. CUMSMICPR03112]>0)
  )
  alors (0)
  sinon (LISTEGEN(TX_ALLOCF;0002))
)
sinon (0)
```



```

)
declarer (HORBASE)
affecter (HORBASE; si ([ AGCPHORAN. BASE] =0)
        alors (
[[[si ( LISTEGEN( TPSAN; CONSTANTE( CONTRAT. CATPERSO) ) =0)
[[[alors (1607)
[[[sinon ( LISTEGEN( TPSAN; CONSTANTE( CONTRAT. CATPERSO) ))
[[[
[[ sinon ([ AGCPHORAN. BASE] )
)
declarer (HEX0)
affecter (HEX0; si (: AD=1)
        alors ([ AD_HEX0. BASE] )
        sinon (0)
)
declarer (PPV)
affecter (PPV; [ P_PPVEXO. MONTANT] +[ P_PPVNONEXO. MONTANT] )
declarer (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( TPEMONTANT;
      : ANNEE;
      : MOIS;
      CONSTANTE( GENERAL. FNAL010) ;
      : NBHEURES;
      : NBJOURS;
      : HSUP;
      [ AGCPITREMU. MONTANT] ;
      [ AGCP_NUMHREMU. MONTANT] ;
      [ B_PXHEURE. MONTANT] ;
      : HORBASE;
      : AD;
      : HEX0;

```


Version 12 Alloc fam Aide à domicile

Pour les heures non exonérées :

- L'allocation familiale taux plein et le complément maladie s'applique pour la part des heures non exonérées
- L'allocation familiale taux réduit et pas de complément maladie pour la part des heures exonérées

Pour cela, modification de la rubrique ALLOCFAMREDUIT.TXEMP

```
si (([ PLAFONDSECU. NUMCAS ] =2) et ( CONSTANTE( CONTRAT. ANNEE_CLOTURE ) <2026) )
alors (
    si (
        (( [ V_NEWAGCP. CUMREMUAF ] ) >
([ V_NEWAGCP. TXSMICMAL ] * [ V_NEWAGCP. CUMSMICPR03112 ] ) )
        et ( [ V_NEWAGCP. CUMREMUAF ] >0)
        et ( [ V_NEWAGCP. CUMSMICPR03112 ] >0)
    )
    alors ( 0)
    sinon ( LISTEGEN( TX_ALLOCF; 0002) )
)
sinon ( si ( CONSTANTE( CONTRAT. TYPESALARIE ) =5)
alors ( LISTEGEN( TX_ALLOCF; 0002) )
sinon ( 0)
)
```

D'autre part il faut modifier la base de cotisation pour le régime AD :

Alias	Désignation	DSN	Base	Taux Salarié	Taux Employe...	Code CTP	CTP1	CTP2	Qualifiant	Base Ass.	+Ass	Comp.Ba...	Code Exo.	Code Cot
☐ Sécurité sociale														
☐ Sur la totalité du Brut														
ACCIDENT	Accident du travail	<input checked="" type="checkbox"/>	BC_BRUT_URSAFF		(E = mc 2]	999	100		Autre	03	<input type="checkbox"/>			045
ALLOCFAM	Allocation familiale	<input checked="" type="checkbox"/>	BC_BRUT_ADNEXO		(E = mc 2]	430			Autre	03	<input type="checkbox"/>			074
ALLOCFAMREDUIT	Allocation familiale Taux réduit	<input checked="" type="checkbox"/>	BC_BRUT_ADEXD		(E = mc 2]	999	100		Autre	03	<input type="checkbox"/>			074
ALLOCFAMCOMP	Complément allocation familiale bulle...	<input checked="" type="checkbox"/>	BC_BRUT_URSAFF		(E = mc 2]						<input type="checkbox"/>			
MALCOMP25	Complément maladie	<input checked="" type="checkbox"/>	BC_BRUT_ADNEXO		(E = mc 2]	635			Autre	03	<input type="checkbox"/>			907
SYNDICAT	Contribution pour le financement des ...	<input checked="" type="checkbox"/>	BC_BRUT_URSAFF		0.016	027			Autre	03	<input type="checkbox"/>			100

AD_HNEKO	Heures non exonérées aide à domicile	70,00				
AD_HEKO	Heures aide à domicile exonérées	81,67				
BRUT	Brut			2361,78		
Cotisations						
PL_JRSEMPLOI	Nombre de jour de la période d'emploi			31,00		
ACCIDENT	Accident du travail	2361,78			3,75%	-88,57
MALADIE	Maladie	2361,78			7,00%	-165,32
MALCOMP25	Complément maladie	1090,03			6,00%	-65,40
SOLIDARITE	Contribution Solidarité Autonomie	2361,78			0,30%	-7,09
VIEILLESSE	Vieillesse déplaçonné	2361,78	0,40%	-9,45	2,02%	-47,71
VIEILLESSE_TRA	Vieillesse sur tranche A	2361,78	6,90%	-162,96	8,55%	-201,93
ALLOCFAM	Allocation familiale	1090,03			5,25%	-57,23
ALLOCFAMREDUIT	Allocation familiale Taux réduit	1271,75			3,45%	-43,88

Version 13

Modification de REDUCPAT.TXEMP

Avant

```
si ( CONSTANTE( GENERAL. MTREDUCHEURE )=0)
alors (
  si ( CONSTANTE( GENERAL. NBSALARIE0101) <=19)
  alors (1.50)
  sinon (
    si ( CONSTANTE( GENERAL. NBSALARIE0101) <=249)
    alors (0.50)
    sinon (0)
  )
)
sinon ( CONSTANTE( GENERAL. MTREDUCHEURE) )
```

Après

```
si ( CONSTANTE( GENERAL. MTREDUCHEURE )=0)
alors (
  si ( CONSTANTE( GENERAL. NBSALARIE0101) <=19)
  alors (1.50)
  sinon (0.50)
)
sinon ( CONSTANTE( GENERAL. MTREDUCHEURE) )
```

Modification de REDUCRTPAT.TXEMP

Avant

```
si ( CONSTANTE( GENERAL. MTREDUCHEURE )=0)
alors (
  si ( CONSTANTE( GENERAL. NBSALARIE0101) <=19)
  alors (1.50)
  sinon (
    si ( CONSTANTE( GENERAL. NBSALARIE0101) <=249)
```

```
        alors (0.50)
        sinon (0)
    )
)
sinon (CONSTANTE(GENERAL.MTREDUCHEURE))
```

Après

```
si (CONSTANTE(GENERAL.MTREDUCHEURE)=0)
alors (
    si (CONSTANTE(GENERAL.NBSALARIE0101)<=19)
    alors (1.50)
    sinon (0.50)
)
sinon (CONSTANTE(GENERAL.MTREDUCHEURE))
```

Version 14

Pour l'aide à domicile

BC_MALCOMP25 et la ZRR

Avant

```

declarer (REMU)
affecter (REMU; si (CONSTANTE( CONTRAT. TYPESALARIE) =5)
    alors ([ AGCPITREMU. MONTANT] * [ AD_HNEXO. BASE] / ([ AD_HEXO. BASE] + [ AD_HNEXO. BASE] ))
    sinon ([ BC_BRUT_URSAFF. MONTANT])
)
si (CONSTANTE( GENERAL. ANNEE_PAYE) >=2026)
alors (
    :remu
)
sinon (
    si (([ PLAFONDSECU. NUMCAS] =2) et (CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2026))
    alors (
        si (
            (([ V_NEWAGCP. CUMREMUAF] ) > ([ V_NEWAGCP. TXSMICMAL] * [ V_NEWAGCP. CUMSMICPR03112] ))
            et ([ V_NEWAGCP. CUMREMUAF] >0)
            et ([ V_NEWAGCP. CUMSMICPR03112] >0)
        )
        alors (
            f_histoclotsit([ BC_BRUT_URSAFF. MONTANT]) + [ BC_BRUT_URSAFF. MONTANT]
            - f_histoclotsit([ BC_MALCOMP25. MONTANT])
        )
        sinon (
            f_histoclotsit([ BC_MALCOMP25. MONTANT]) * (-1)
        )
    )
    sinon (:remu)
)
)

```

Après

```

declarer (MTAV2025)
affecter (MTAV2025;
    si (
        (([ V_NEWAGCP. CUMREMUAF] ) >
([ V_NEWAGCP. TXSMICMAL] * [ V_NEWAGCP. CUMSMICPR03112] ))
        et ([ V_NEWAGCP. CUMREMUAF] >0)
        et ([ V_NEWAGCP. CUMSMICPR03112] >0)
    )
    alors (
        f_histoclotsit([ BC_BRUT_URSAFF. MONTANT]) +

```

```

[ BC_BRUT_URSAFF. MONTANT]
    - f_histoclotsit([ BC_MALCOMP25. MONTANT])
    )
sinon (
    f_histoclotsit([ BC_MALCOMP25. MONTANT]) *(- 1)
    )
)
si ( CONSTANTE( CONTRAT. TYPESALARIE) =5)
alors ([ BC_BRUT_ADNEX0. MONTANT])
sinon (
    si ([ AGCPEX0. BASE]=1)
    alors (
        : MTAV2025
    )
    sinon (
        si (([ PLAFONDSECU. NUMCAS]=2) et
(CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2026)
        alors ( : MTAV2025)
        sinon ( [ BC_BRUT_URSAFF. MONTANT])
        )
    )
)

```

ALLOCFAM.TXEMP

Avant

Après

```

declarer ( MTAV2025)
affecter ( MTAV2025;
    si (
        (([ V_NEWAGCP. CUMREMUAF] ) >
([ V_NEWAGCP. TXSMICMAL] * [ V_NEWAGCP. CUMSMICPR03112] ))
        et ([ V_NEWAGCP. CUMREMUAF] >0)
        et ([ V_NEWAGCP. CUMSMICPR03112] >0)
    )
    alors ( LISTEGEN( TX_ALLOCF; 0001))
    sinon ( 0)
)

```

```

)
si ( CONSTANTE( CONTRAT. TYPESALARIE) =5)
alors ( LISTEGEN( TX_ALLOCF; 0001) )
sinon (
    si ([ AGCPEX0. BASE] =1)
    alors (
        : MTAV2025
    )
    sinon (
        si (([ PLAFONDSECU. NUMCAS] =2) et
(CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2026))
        alors ( : MTAV2025)
        sinon ( LISTEGEN( TX_ALLOCF; 0001) )
    )
)
)

```

ALLOCFAMREDUIT.TXEMP

Avant

```

si (([ PLAFONDSECU. NUMCAS] =2) et ( CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2026))
alors (
    si (
        (([ V_NEWAGCP. CUMREMUAF] ) >
([ V_NEWAGCP. TXSMICMAL] *[ V_NEWAGCP. CUMSMICPRO3112]))
        et ([ V_NEWAGCP. CUMREMUAF] >0)
        et ([ V_NEWAGCP. CUMSMICPRO3112] >0)
    )
    alors ( 0)
    sinon ( LISTEGEN( TX_ALLOCF; 0002) )
)
sinon ( 0)

```

Après

```

declarer ( MTAV2025)
affecter ( MTAV2025;
    si (
        (([ V_NEWAGCP. CUMREMUAF] ) >

```

```
([ V_NEWAGCP. TXSMICMAL] *[ V_NEWAGCP. CUMSMICPRO3112] ) )
    et ([ V_NEWAGCP. CUMREMUAF] >0)
    et ([ V_NEWAGCP. CUMSMICPRO3112] >0)
    )
    alors ( 0)
    sinon (LISTEGEN( TX_ALLOCF; 0002))
)
si ([ AGCPEX0. BASE]=1)
alors (
    si (CONSTANTE( CONTRAT. TYPESALARIE)=5)
    alors (
        LISTEGEN( TX_ALLOCF; 0002)
    )
    sinon (
        :MTAV2025
    )
)
sinon (
    si (([ PLAFONDSECU. NUMCAS]=2) et (CONSTANTE( CONTRAT. ANNEE_CLOTURE) <2026) )
    alors ( :MTAV2025)
    sinon ( 0)
)
)
```

Version 15 Correctif

Fonction F66AF

Alerte Christophe

dans la fonction F_66AF il y a une connerie avec la prise en compte de 66_AFACCINTER.

ça remplace le calcul des part enfant alors que ça ne dois pas.

Avant

```
si ( ( CONSTANTE(CONTRAT. NATURECONTRAT)=3) et ([ NUMCONV. NUMCONV]=66) et
([ 66_AFN BENF. MONTANT] >=NUMERO) )
alors (
  si ( REMU=1)
  alors (
    si ( CONV=1)
    alors (
      /* Rémunération conventionnelle*/
      si ([ 66_AFACCINTER. MONTANT]=0)
      alors (
        si ( NBJACCUEIL<16)
        alors ( NBJACCUEIL/26)
        sinon (1)
        *
        si ( NUMERO=1)
        alors ( /* Enfant 1 */
          ( PROG+PRECPR0)
          )
        sinon ( /* Enfants suivant */
          ( PROG- PRECPR0)
          )
        *[ VP. VP]
        *( 1+CONSTANTE( GENERAL. TAUXSUJETION66) /100)
        )
      sinon (
```

```

        NBJACCUEIL*5.06*CONSTANTE( GENERAL. SMICHOR)
    )
)
sinon (
/*Rémunération légale*/
    si ([ 66_AFACCINTER. MONTANT]=0)
    alors (
        si (NUMERO=1)
        alors ( /* Enfant 1 */
            NBJACCUEIL
            /CONSTANTE( GENERAL. NBJMOISPAYE)
            *CONSTANTE( GENERAL. SMICMENS)
        )
        sinon ( /* Enfants suivant */
            NBJACCUEIL
            /CONSTANTE( GENERAL. NBJMOISPAYE)
            *70
            *CONSTANTE( GENERAL. SMICHOR)
        )
    )
    sinon (0)
)
)
sinon (
    si (CONV=1)
    /* Absence conventionnelle*/
    alors (
        si ( (NBJACCUEIL<>0) et
(NBJACCUEIL<CONSTANTE( GENERAL. NBJMOISPAYE))
            et (NBJACCUEIL<16)
        )
        alors (
            si (NUMERO=1)
            alors ( /* Enfant 1 */
                maximum( 0; ( CONSTANTE( CONTRAT. CIVJOURSTRAVAIL_ EMB) - NBJACCUEIL -
(NBJABSCP)))
                /CONSTANTE( GENERAL. NBJMOISPAYE)
                *0.8
                *CONSTANTE( GENERAL. SMICMENS)
            )

```

```

        sinon ( /* Enfants suivant */
            maximum( 0; ( CONSTANCE( CONTRAT. CIVJOURSTRAVAIL_EMB) - NBJACCUEIL-
(NBJABSCP)))

            /CONSTANCE( GENERAL. NBJMOISPAYE)
            *0.8*70
            *CONSTANCE( GENERAL. SMICHOR)
        )
    )
    sinon ( 0)
)
sinon (
/* Absence légale */
    si ( NUMERO=1)
        alors ( /* Enfant 1 */
            maximum( 0; ( CONSTANCE( CONTRAT. CIVJOURSTRAVAIL_EMB) - NBJACCUEIL-
(NBJABSCP)))

            /CONSTANCE( GENERAL. NBJMOISPAYE)
            *0.8
            *CONSTANCE( GENERAL. SMICMENS)
        )
        sinon (
            maximum( 0; ( CONSTANCE( CONTRAT. CIVJOURSTRAVAIL_EMB) - NBJACCUEIL-
(NBJABSCP)))

            /CONSTANCE( GENERAL. NBJMOISPAYE)
            *0.8*70
            *CONSTANCE( GENERAL. SMICHOR)
        )
    )
)
)
sinon ( 0)

```

Après

```

si ( ( CONSTANCE( CONTRAT. NATURECONTRAT)=3) et ([ NUMCONV. NUMCONV]=66) et
([ 66_AFN BENF. MONTANT] >=NUMERO) )
alors (
    si ( REMU=1)
        alors (

```

```

si ( CONV=1)
alors (
/* Rémunération conventionnelle*/
    si ( NBJACCUEIL<16)
    alors ( NBJACCUEIL/26)
    sinon ( 1)
    *
    si ( NUMERO=1)
    alors ( /* Enfant 1 */
        ( PROG+PRECPR0)
        )
    sinon ( /* Enfants suivant */
        ( PROG- PRECPR0)
        )
    *[ VP. VP]
    *(1+CONSTANTE( GENERAL. TAUXSUJETION66) /100)
    )
sinon (
/*Rémunération légale*/
    si ( NUMERO=1)
    alors ( /* Enfant 1 */
        NBJACCUEIL
        /CONSTANTE( GENERAL. NBJMOISPAYE)
        *CONSTANTE( GENERAL. SMICMENS)
        )
    sinon ( /* Enfants suivant */
        NBJACCUEIL
        /CONSTANTE( GENERAL. NBJMOISPAYE)
        *70
        *CONSTANTE( GENERAL. SMICHOR)
        )
    )
)
sinon (
    si ( CONV=1)
    /* Absence conventionnelle*/
    alors (
        si ( ( NBJACCUEIL<>0) et
(NBJACCUEIL<CONSTANTE( GENERAL. NBJMOISPAYE) )
            et ( NBJACCUEIL<16)
            )

```

```

alors (
  si (NUMERO=1)
  alors ( /* Enfant 1 */
    maximum( 0; ( CONSTANTE( CONTRAT. CIVJOURSTRAVAIL_EMB) - NBJACCUEIL-
(NBJABSCP)))

    /CONSTANTE( GENERAL. NBJMOISPAYE)
    *0.8
    *CONSTANTE( GENERAL. SMICMENS)
    )
  sinon ( /* Enfants suivant */
    maximum( 0; ( CONSTANTE( CONTRAT. CIVJOURSTRAVAIL_EMB) - NBJACCUEIL-
(NBJABSCP)))

    /CONSTANTE( GENERAL. NBJMOISPAYE)
    *0.8*70
    *CONSTANTE( GENERAL. SMICHOR)
    )
  )
  sinon (0)
)
sinon (
/* Absence légale */
  si (NUMERO=1)
  alors ( /* Enfant 1 */
    maximum( 0; ( CONSTANTE( CONTRAT. CIVJOURSTRAVAIL_EMB) - NBJACCUEIL-
(NBJABSCP)))

    /CONSTANTE( GENERAL. NBJMOISPAYE)
    *0.8
    *CONSTANTE( GENERAL. SMICMENS)
    )
  sinon (
    maximum( 0; ( CONSTANTE( CONTRAT. CIVJOURSTRAVAIL_EMB) - NBJACCUEIL-
(NBJABSCP)))

    /CONSTANTE( GENERAL. NBJMOISPAYE)
    *0.8*70
    *CONSTANTE( GENERAL. SMICHOR)
    )
  )
)
)
sinon (0)

```