

Pbm 5

1 Réécriture de la fonction :

Fonction $z(m)$

Si $m > 0$ alors retourner $m-1$;

Sinon

$R1 = z(m+2)$ [appel non terminal]

Retourner $z(R1)$; [appel terminal]

Fin

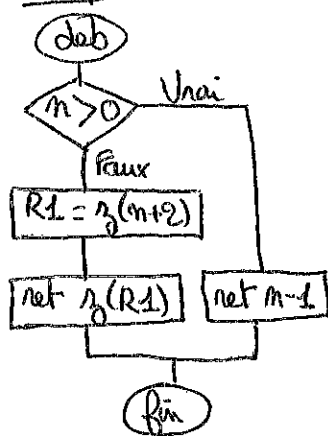
Fait

$$① z(4) = z(z(-2)) = z(0) = 0$$

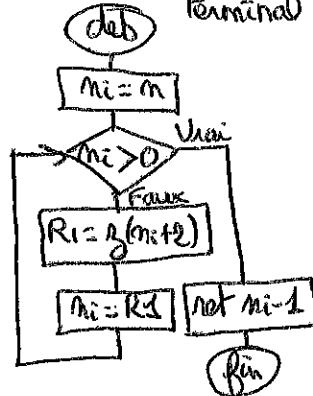
$$\rightarrow z(z(0)) = z(0) = 0$$

$$\rightarrow z(z(2)) = z(-1) = 0 \rightarrow \text{à mieux développer par la suppress° de l'appel non terminal}$$

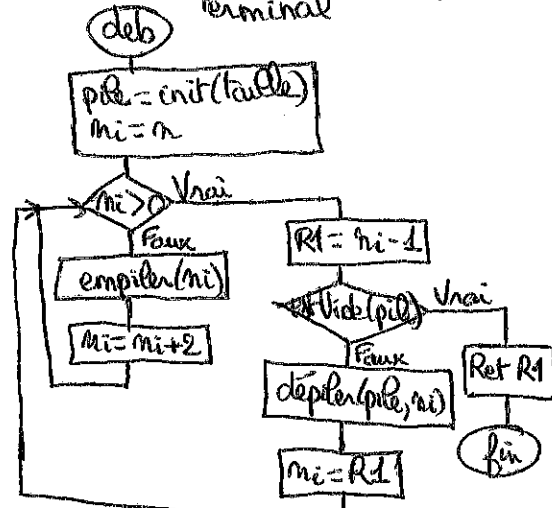
② Etape 0



Etape 1: Suppression de l'appel terminal



Etape 2: Suppression de l'appel non terminal



1 Programme itératif

Fonction $z(m)$

$pile = \text{init}(taille);$

$mi = m;$

$boolFin = \text{F}$

Tant que NON boolFin

Tant que $mi \leq 0$

$\text{empiler}(pile, mi);$

$mi = mi + 2;$

Fait

$R1 = mi - 1;$

Si NON estVide(pile) alors

$\text{dépiler}(pile, mi)$

$mi = R1;$

Sinon

$boolFin = \text{Vrai}$

Fin

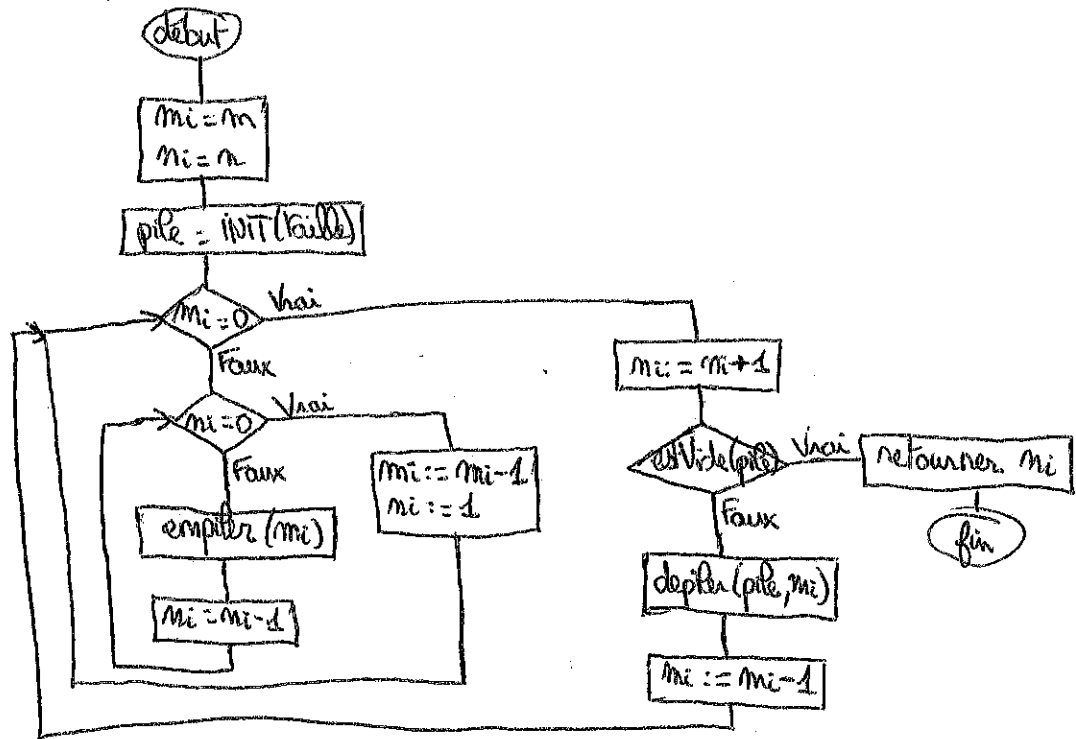
Fait
Retourner $R1$

Fait

Eg: Ici on aurait pu pas besoin d'une pile, puisque mi est it de suite mis à $R1$.

Il suffirait de d'utiliser 1 compteur, qui s'incrémente et se décrémente, pour avoir le nb d'appels récursifs.

1. Suppression de l'appel non terminal



2. Programme itératif

Fonction HORS(m, n)

mi(mi) := m;

ni(ni) := n;

m(pile) := initPile(taille);

m(finPile) := Faux;

Tant que HORS(m(finPile))

| Si em(mi) ≠ 0 alors [Tant que possible aussi]

| | Tant que em(mi) ≠ 0

| | | empiler(pile, em(mi));

| | | m(mi) := em(mi) - 1;

| | Fin

| | m(mi) := em(mi) - 1;

| | m(mi) := 1;

| Sinon [em(mi) = 0]

| | m(mi) := em(mi) + 1;

| | Si HORS(estVide(pile)) alors

| | | depiler(pile, m(mi));

| | | m(mi) := em(mi) - 1;

| | Sinon

| | | m(finPile) := Vrai;

| | Fin

Fin
Fin
retourner em(mi);