Cruz Pérez

FIC/UNACH

PROGRAMACIÓN

FINAL

Calculo del Área de Acero en Zapatas Rígidas

Las zapatas aisladas, son elementos estructurales de concreto armado, que sirven para repartir las cargas de la columna al suelo, de tal manera que la resistencia del suelo las soporte. Se deduce que suelos de buena resistencia tendrán zapatas de menor dimensión, con respecto a las construidas en suelos de menor resistencia.2. Su diseño sirve de base para otro tipo de cimentaciones. Los otros tipos de cimientos fallan por mecanismos similares a los de éstas zapatas: por flexión, adherencia y anclaje, cortante punzonante y cortante por flexión.3. El diseño consiste en calcular, la forma y dimensiones del concreto, así como la cantidad y tipos de acero de la zapata.4. Se necesita como datos, conocer: la carga axial de la superestructura, la sección y aceros de la columna que soporta, y la resistencia admisible del suelo (*q adm*), sobre el que se diseña la zapata.

Elementos para el diseño de zapata aislada.

ELEMENTOS BASICOS

:A, B = Dimensiones en planta de la zapatas,

t = Dimensiones en planta de la columna

m = Longitud del volado de la zapata

H = peralte de la zapata

P = carga axial actuante

qadm = capacidad de carga admisible del suelo

Ld = longitud de anclaje por compresión (o tracción) delacero de columna

*g*= Peso específico promedio del relleno

Df = profundidad de cimentación

s/c piso = sobrecarga de piso = 500 kg/m

Hay que encontrar el esfuerzo neto (q neto) que soporta el suelo:q neto = qadm -

*g*\* Df - s/c piso6. Hay que calcular el peso total

*Pt*

de la superestructura que llega al suelo, incluyendo el peso propio de zapata:

Se va a encontrar la proporción n entre el peso de zapata *Pz* y la carga de servicio *P*, como función del esfuerzo neto:

De *n = Pz / P, P + Pz = q neto x A*, y

*Pz =*γ*c\* A \* B \* H*

Siendo:-

γc= Peso volumétrico del concreto armado.

A, B, H = dimensiones en planta y elevación de la zapata.-q neto = esfuerzo neto Se obtiene:

1\*1−=*H cqneton*γ

P de zapata = n x P de servicio

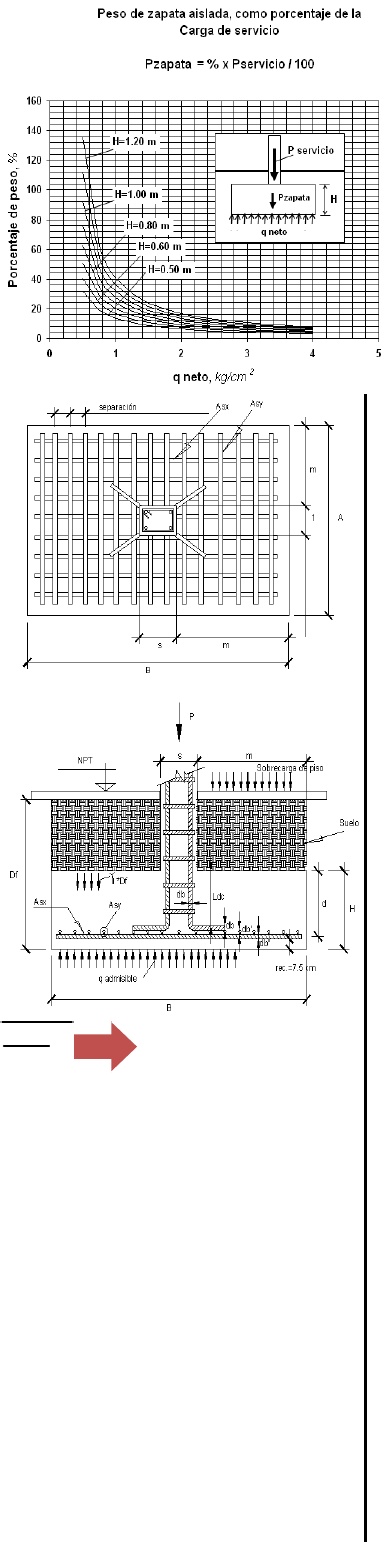
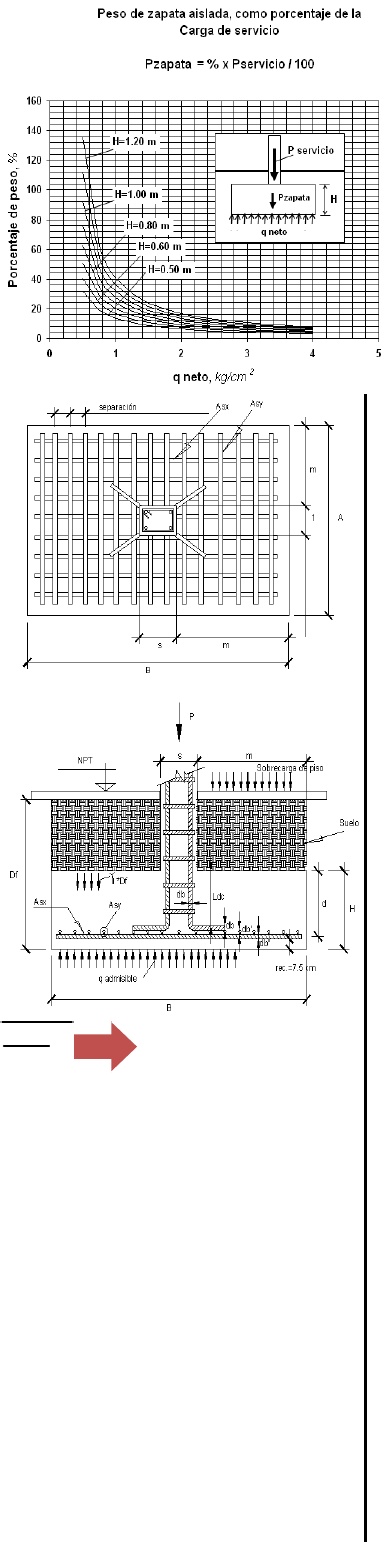
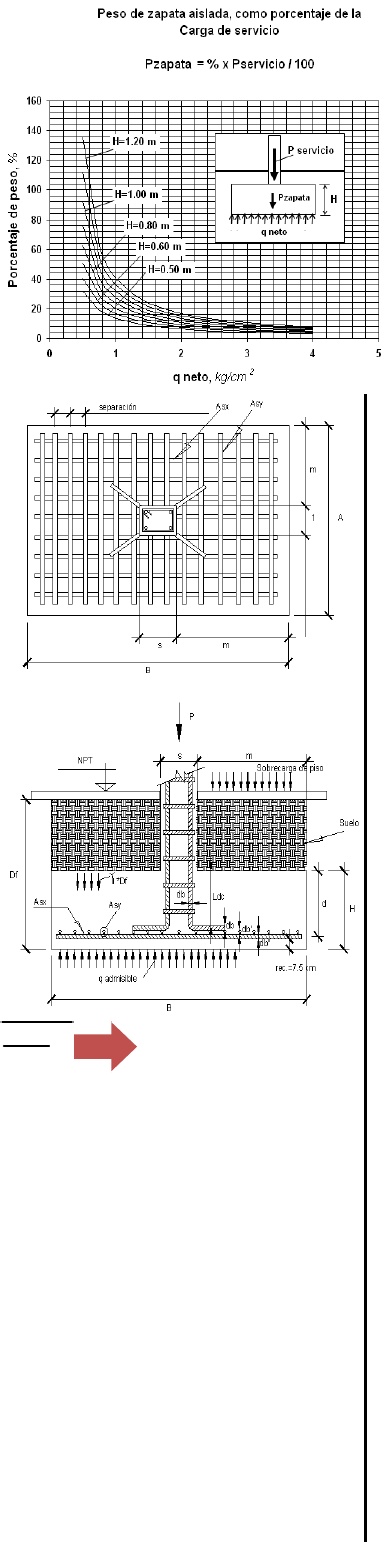
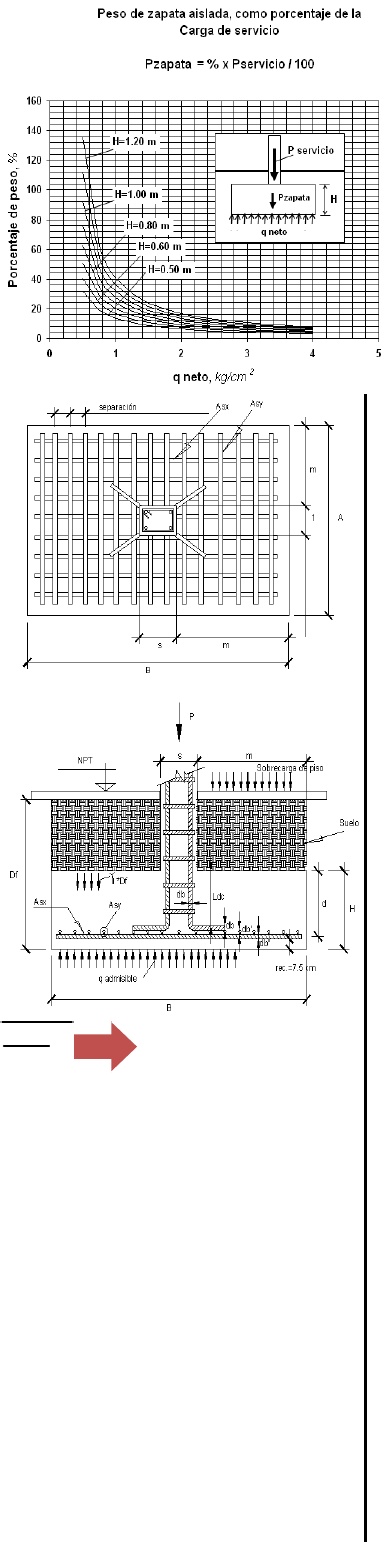
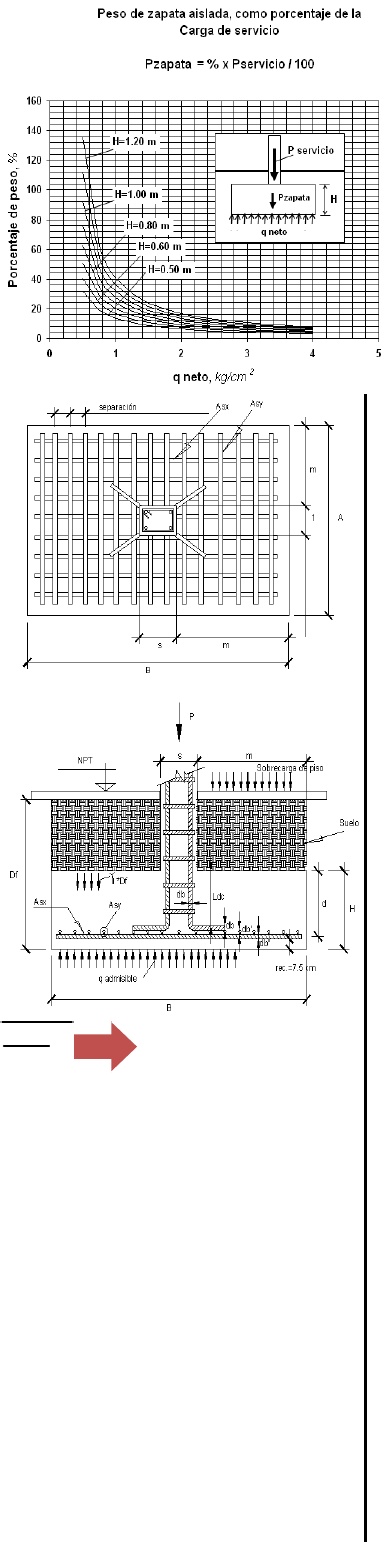
Fig. 2. Gráfica para pre-dimensionado de zapata aislada. Se suele usar:

Pt = P + (%) P, el %P se obtiene de la Fig.

Determinamos el área de zapata requerida:

A zapata = (Pt) /q neto8.

Como se busca que en ambos sentidos la zapata tenga el mismo volado:(s + 2m)(t + 2m ) = A zapata



 Resolviendo la ecuación se obtiene m aproximadamente: \_\_\_\_ m = (√Azap / 2) - (s + t)/49.

Luego las dimensiones de A y B son:A = 2m + tB = 2m + s \_\_\_\_\_\_\_\_\_

A =√(A zapata) – (s-t)/2 \_\_\_\_\_\_\_\_\_

B =√(A zapata) + (s-t)/2

Luego dimensionamos el peralte

*H*:*H*

se calcula cuando se determine el peralte efectivo "d", mediante la verificación por:

-Longitud de desarrollo-Cortante por punzonamiento-Cortante por flexión

11. La longitud de desarrollo a compresión está dada por:

 \_\_ ld = 0.08 \* fy \* db /√f'c …

Norma ACI-318ld = 0.004 db \* fy, old = 20 cm, el que sea mayor. db = diámetro de la varilla de la columna

db'= diámetro de la varilla superior de la parrilla

db"= diámetro de la varilla inferior de la parrilla

Por tanto H deberá ser igual a:H = ld + db` + db” + recubrimiento.

Elementos que componen el peralte de la zapata por longitud de desarrollo.

Hay que calcular la reacción última (qu) del suelo:Pu = 1.5 D + 1.8 L

Pu = 1.2D + 1.6 L (Normas ACI-318)

qu = Pu/(A\*B)

El esfuerzo cortante por punzonamiento se calcula con:

Vp = Vu – 2\* (s+d)(t+d)\*d-

v actuante = Vp / (perímetro \* d)

v actuante = qu\* [ A\*B - (s+d)\*(t + d)] / [2d\*(s + t + 2\*d)]....(A)

Falla por punzonamiento. Ensayo en la UNPRG.Lambayeque. Perú. por punzonamiento y bloque equivalente.

El que tendrá que ser menor o igual que el esfuerzo cortante admisible: \_\_ v admisible =

φ\* 0.27(2 + 4/ß)√f'c

…Norma ACI 318

ß = s/t (lado mayor a lado menor de columna)o también: \_\_ v admisible =

φ\* 1.1√f'c ,

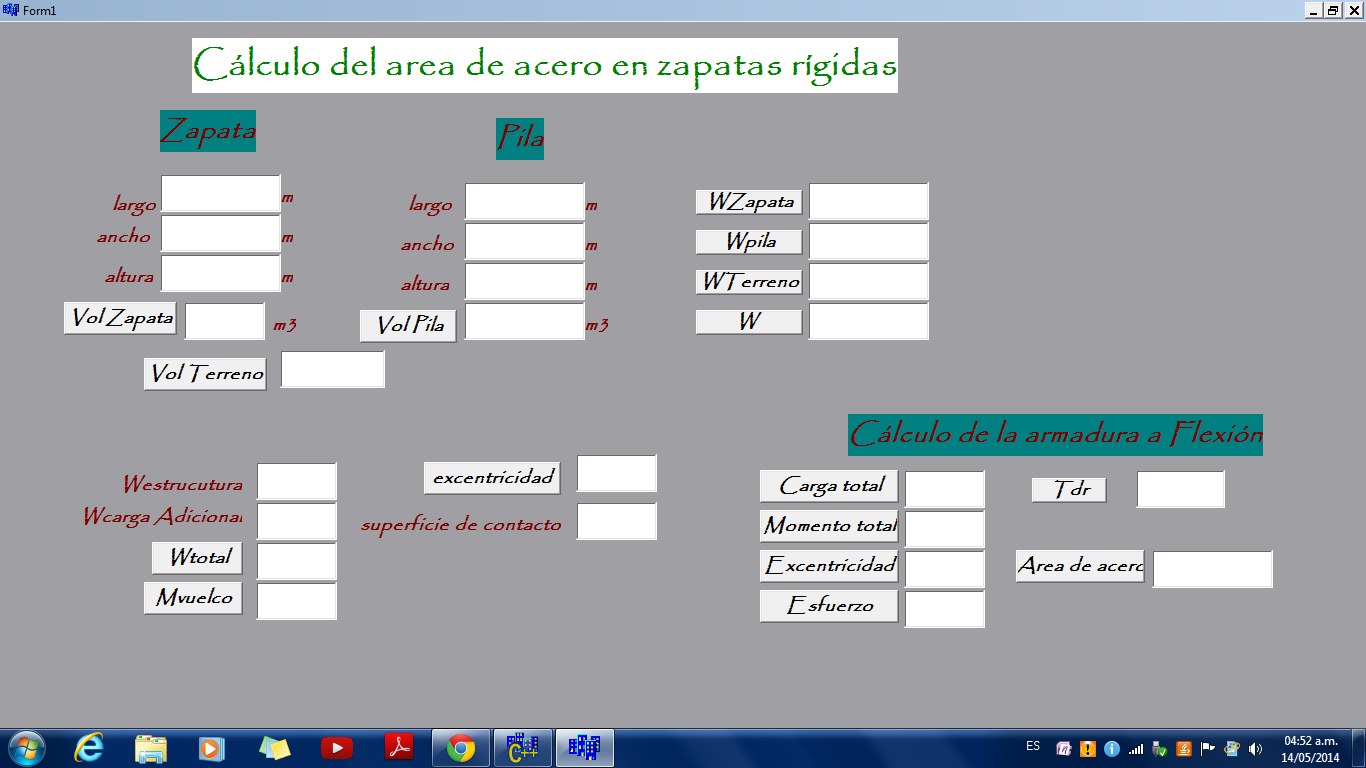
φ= 0.85 …Norma ACI 318...

(B)El que sea menor.15.

"d2" se obtiene al igualar las expresiones(A) = (B)qu\* [ A\*B - (s+d)\*(t + d)] / [2d\*(s + t + 2\*d)] =φ\* 0.27(2 + 4/ß)√f'có {φ\* 1.1

√f'c

Interfaz Grafica



Programación

/---------------------------------------------------------------------------

#include <vcl.h>

#pragma hdrstop

#include "Unit1.h"

//---------------------------------------------------------------------------

#pragma package(smart\_init)

#pragma resource "\*.dfm"

double Lz, Az, Hz, Lp,Ap,Hp, Dhorm, DTerr, g, Vz, Vp, Vt, Wz, Wp, Wt, Wtt,West, Wsc, Wver ;

TForm1 \*Form1;

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button2Click(TObject \*Sender)

{

Lp= Edit4->Text.ToDouble();

Ap= Edit5->Text.ToDouble();

Hp= Edit6->Text.ToDouble();

Dhorm= 2500;

DTerr= 1800;

g=9.8;

Vp= Lp\*Ap\*Hp;

Edit8->Text=AnsiString(Vp);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

Lz= Edit1->Text.ToDouble();

Az= Edit2->Text.ToDouble();

Hz= Edit3->Text.ToDouble();

Vz= Lz\*Az\*Hz;

Edit7->Text=AnsiString(Vz);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button3Click(TObject \*Sender)

{

Vt=((Lz\*Az)-(Lp\*Ap))\*Hz;

Edit9->Text=AnsiString(Vt);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button4Click(TObject \*Sender)

{

Wz=(Dhorm\* Vz\*g) ;

Edit10->Text=AnsiString(Wz);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button5Click(TObject \*Sender)

{

Wp=(Dhorm\* Vp\*g) ;

Edit11->Text=AnsiString(Wp);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button6Click(TObject \*Sender)

{

Wt=(DTerr\* Vt\*g) ;

Edit12->Text=AnsiString(Wt);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button7Click(TObject \*Sender)

{

Wtt= Wz+Wp+Wt;

Edit13->Text=AnsiString(Wtt);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button8Click(TObject \*Sender)

{

West=Edit14->Text.ToDouble();

Wsc=Edit15->Text.ToDouble();

Wver= Wtt+Wsc;

Edit16->Text=AnsiString(Wver);

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button13Click(TObject \*Sender)

{

if (C==1)

{

A=Edit1->Text.ToDouble();

Xi=Edit7->Text.ToDouble();

Yi=Edit6->Text.ToDouble();

s=Edit29->Text.ToDouble();

k11= (A\*Xi\*Xi);

Co=(log(Yi))-k11;

if(Co==0)

{

Ysol=(exp(A\*k11));

Edit31->Text=AnsiString(Ysol);

}

if(Co=!0)

{

Ysol=(exp((A\*k11)+Co));

Edit31->Text=AnsiString(Ysol);

}

}

if (C==2)

{

A=Edit2->Text.ToDouble();

B=Edit3->Text.ToDouble();

Xi=Edit7->Text.ToDouble();

Yi=Edit6->Text.ToDouble();

s=Edit29->Text.ToDouble();

Ex= (exp((A\*Xi)/A));

B12=B/2;

Ce=((Yi\*Yi)\*B12)+Yi-Ex;

Ce1=(Ex+Ce)\*-1;

F1=(-1+(sqrt(1-(4\*B12\*Ce1))))/(2\*B12);

F2=(-1-(sqrt(1-(4\*B12\*Ce1))))/(2\*B12);

Edit31->Text=AnsiString(F1);

Edit32->Text=AnsiString(F2);

}

if (C==3)

{

A=Edit4->Text.ToDouble();

B=Edit5->Text.ToDouble();

Xi=Edit7->Text.ToDouble();

Yi=Edit6->Text.ToDouble();

s=Edit29->Text.ToDouble();

A12=(A/2) ;

A22=(Xi\*Xi\*A12);

B22=B/2;

C1c=((Yi\*Yi)\*B22)+Yi-A22;

C2c=(A22+C1c)\*-1;

FA1=(-1+(sqrt(1-(4\*B22\*C2c))))/(2\*B22);

FA2=(-1-(sqrt(1-(4\*B22\*C2c))))/(2\*B22);

Edit31->Text=AnsiString(FA1);

Edit32->Text=AnsiString(FA2);

}

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button15Click(TObject \*Sender)

{

if (C==1)

{

EA1=Edit30->Text.ToDouble();

ER2=Edit31->Text.ToDouble();

VE= ((ER2-EA1)/EA2)\*100;

Edit33->Text=AnsiString(VE);

}

if (C==2)

{

EA1=Edit30->Text.ToDouble();

ER1=Edit31->Text.ToDouble();

ER2=Edit32->Text.ToDouble();

VE1= ((ER1-EA1)/ER1)\*100;

VE2= ((ER2-EA1)/ER2)\*100;

Edit33->Text=AnsiString(VE1);

Edit34->Text=AnsiString(VE2);

}

if (C==3)

{

EA1=Edit30->Text.ToDouble();

ER1=Edit31->Text.ToDouble();

ER2=Edit32->Text.ToDouble();

VE1= ((ER1-EA1)/ER1)\*100;

VE2= ((ER2-EA1)/ER2)\*100;

Edit33->Text=AnsiString(VE1);

Edit34->Text=AnsiString(VE2);

}

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button14Click(TObject \*Sender)

{

Edit1->Text="";

Edit2->Text="";

Edit3->Text="";

Edit4->Text="";

Edit5->Text="";

Edit6->Text="";

Edit7->Text="";

Edit8->Text="";

Edit9->Text="";

Edit10->Text="";

Edit11->Text="";

Edit12->Text="";

Edit13->Text="";

Edit14->Text="";

Edit15->Text="";

Edit16->Text="";

Edit17->Text="";

Edit18->Text="";

Edit19->Text="";

Edit20->Text="";

Edit21->Text="";

Edit22->Text="";

Edit23->Text="";

Edit24->Text="";

Edit25->Text="";

Edit26->Text="";

Edit27->Text="";

Edit28->Text="";

Edit29->Text="";

Edit30->Text="";

Edit31->Text="";

Edit32->Text="";

Edit33->Text="";

Edit34->Text="";

Edit1->SetFocus();

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button12Click(TObject \*Sender)

{

Close();

}

//---------------------------------------------------------------------------