

ДОДАТОК А



**ДЕРЖАВНИЙ КОНЦЕРН «УКРОБОРОНПРОМ»
ДЕРЖАВНЕ ПІДПРИЄМСТВО
«СУДНОБУДІВНИЙ ЗАВОД ІМЕНІ 61 КОМУНАРА»**

Вул. Адміральська, 38 м Миколаїв 54001;

Телефон (+380512) 670093 факс (+380512) 670093

Ідентифікаційний код за ЄДРПОУ 14313240

URL: www.shipyard61.com.ua

E- mail: info@shipyard61.com.ua

Д О В І Д К А П Р О В П Р О В А Д Ж Е Н Н Я

Видана **Кононову Максиму Анатолійовичу** у тому, що матеріали його бакалаврської роботи **«Програмний комплекс підтримки інженерних розрахунків у машинобудуванні»** (керівник к. т. н., доц. Романкевич В. О. плануються до впровадження як самостійний програмний продукт «EngineeringCalc 2.0».

Розроблений комплекс успішно використовується та тестується на підприємстві. Завдяки даній програмі були суттєво зменшені часові витрати на розрахунок необхідної кількості різних видів металопрокату і тросу, розривного зусилля та діаметра канатів, збільшена точність обчислень, отримана можливість представлення результатів високоточних інженерних розрахунків у різних системах числення з метою їх шифрування. Крім того використання програми сприяє більш швидкому документуванню потреб у різноманітних матеріалах. Калькулятор може бути вдосконалений та доповнений новими функціональними можливостями в робочому порядку з урахуванням виробничих потреб.

Головний інженер

А. Р. Ярошевич

ДОДАТОК Б

Лістинг програми

```
var Form1: TForm1; s,t1:string;
c,comma:char; pr,i:byte; set1:set of char; b:boolean;
s1,s2{исходные строки},od{остаток от деления},
se1,se2{выровненные строки}:string;
sys{система счисления},sys1,sys2,
prec{точность деления}:byte;
z1,z2{позиции запятых}:integer;
{арифм. операции}op_plusb,op_minusb,
{поля ввода}b_ae,b_quantity,b_thickness,
b_kgm,b_rope_effort,b_divprec,converter:boolean;
implementation
{$R *.lfm}
//=====Арифметические операции=====
//преобразование цифры в символ
function ntoch(i:byte):string;
begin
if i>9 then begin i:=i+55; ntoch:=chr(i); end
else ntoch:=inttostr(i); end;
//преобразование символа в цифру
function chton(ch:char):byte;
var set1:set of char; i:byte;
begin set1:=[];
for i:=48 to 57 do
set1:=set1+[chr(i)];
if ch in set1 then chton:= strtoint(ch)
else chton:=ord(ch)-55; end;
//выравнивание чисел с запятой
function normalize(var s1,s2:string):byte;
var i:byte;
begin
z1:=pos(comma,s1); z2:=pos(comma,s2);
if z1=0 then z1:=length(s1)+1;
if z2=0 then z2:=length(s2)+1;
if z1>z2 then normalize:=z1 else normalize:=z2;
if z1>z2 then for i:=0 to z1-z2 do s2:='0'+s2
else if z1<z2 then for i:=0 to z2-z1 do s1:='0'+s1;
if pos(comma,s1)>0 then z1:=length(s1)-pos(comma,s1) else z1:=0;
if pos(comma,s2)>0 then z2:=length(s2)-pos(comma,s2) else z2:=0;
if z1>z2 then for i:=1 to z1-z2 do s2:=s2+'0'
else if z1<z2 then for i:=1 to z2-z1 do s1:=s1+'0';
if pos(comma,s1)>0 then delete(s1,pos(comma,s1),1);
if pos(comma,s2)>0 then delete(s2,pos(comma,s2),1); end;
//выравнивание чисел без запятой
procedure eqsize(var s1,s2:string);
var i:byte;
begin
if length(s1)<length(s2) then
for i:=1 to length(s2)-length(s1) do s1:='0'+s1;
```

```

if length(s1)>length(s2) then
for i:=1 to length(s1)-length(s2) do s2:='0'+s2;
se1:=s1; se2:=s2; end;
//удаление ноля в начале строки
function zerodelb(s:string):string;
var i:integer;
begin
if s="" then exit; i:=1;
if (s[1]='0') and (pos(comma,s)>0) then exit;
while s[i]='0' do inc(i);
delete(s,1,i-1);
if length(s)=0 then s:='0';
zerodelb:=s; end;
//удаление ноля в конце строки
function zerodele(s:string):string;
var i:integer;
begin i:=length(s){-1};
while s[i]='0' do dec(i);
delete(s,i+1,length(s)-i);
if length(s)=0 then s:='0';
zerodele:=s; end;
//сравнение чисел
function compare(s1,s2:string):char;
var i:byte;
begin
if s1=s2 then compare:='=' else begin
normalize(s1,s2); i:=1;
while s1[i]=s2[i] do inc(i);
if ord(s1[i])<ord(s2[i]) then compare:='<' else compare:='>';
end; end;
//проверка соответствия числа указанной системе счисления
function check(s:string):boolean;
var i:byte; myset:set of char; t:string;
begin myset:=[];
if (sys>36) or (sys<2) or (s='') then begin
check:=false; exit; end;
myset:=set1+[comma]+'.';
for i:=48 to 48+sys-1 do
myset:=myset+[chr(i)];
if sys>10 then
for i:=65 to 65+sys-9 do
myset:=myset+[chr(i)]; i:=1;
for i:=1 to length(s) do
if not(s[i] in myset) then begin check:=false; exit; end
else check:=true; end;
//обработка знака при умножении и делении
function sign(var s1,s2:string):boolean;
begin
if (s1[1]='-') and (s2[1]='-') then begin
sign:=false; delete(s1,1,1); delete(s2,1,1); exit; end;
if (s1[1]='-') and (s2[1]<>'-') then begin
sign:=true; delete(s1,1,1); exit; end;

```

```

if (s1[1]<>'-' and (s2[1]='-') then begin
sign:=true; delete(s2,1,1); exit; end; end;
//сложение чисел без знака
function op_plus(s1,s2:string):string;
var e:string; c,d,i,r,p,q,cp: byte;
begin r:=0; e:=""; cp:=0;
if pos(comma,s1)>0 then z1:=length(s1)-pos(comma,s1) else z1:=0;
if pos(comma,s2)>0 then z2:=length(s2)-pos(comma,s2) else z2:=0;
if z1>z2 then cp:=z1 else cp:=z2;
if op_plusb=false then
if (pos(comma,s1)>0) or (pos(comma,s2)>0) then
{cp:=} normalize(s1,s2);
if length(s1)<>length(s2) then eqsize(s1,s2);
p:=length(s1); q:=length(s2); i:=0; r:=0;
while (p>0) and (q>0) do begin
c:=chton(s1[p]); dec(p);
d:=chton(s2[q]); dec(q); r:=c+d+i;
if r>sys-1 then begin r:=r-sys; i:=1 end else i:=0;
e:=ntoch(r)+e; end; if i=1 then e:='1'+e;
e:=zerodelb(e);
if op_plusb=false then begin
if cp>0 then insert(comma,e,length(e)-cp+1);
if pos(comma,e)=1 then e:='0'+e;
if pos(comma,e)=length(e) then delete(e,pos(comma,e),1);
if pos(comma,e)=0 then e:=zerodelb(e) else begin
e:=zerodele(e); e:=zerodelb(e); end;
if pos(comma,e)=length(e) then delete(e,length(e),1); end;
op_plus:=e; end;
//отнимание чисел без знака
function op_minus(s1,s2:string):string;
var s,t:string; n,k,p,i,r:integer;
cp:byte; ms:boolean;
begin ms:=false; cp:=0;
if op_minusb=false then
if compare(s1,s2)='<' then begin
t:=s1; s1:=s2; s2:=t; ms:=true; end;
if (pos(comma,s1)>0) or (pos(comma,s2)>0) then begin
if pos(comma,s1)>0 then z1:=length(s1)-pos(comma,s1) else z1:=0;
if pos(comma,s2)>0 then z2:=length(s2)-pos(comma,s2) else z2:=0;
if z1>=z2 then cp:=z1 else cp:=z2; normalize(s1,s2); end;
if length(s1)<>length(s2) then eqsize(s1,s2);
n:=length(s1); r:=0; i:=0;
while (n>0) do begin
k:=chton(s1[n])-i; p:=chton(s2[n]); dec(n);
if k=-1 then k:=sys-1 else
if k<p then begin k:=k+sys; i:=1; end
else i:=0; r:=k-p;
s:=ntoch(r)+s; end;
if (pos('0',s)=1) and (cp=0) then s:=zerodelb(s);
if op_minusb=false then begin
if cp>0 then insert(comma,s,length(s)-cp+1);
if converter=false then

```

```

if s[1]='0' then while s[1]='0' do delete(s,1,1);
i:=1;
if pos(comma,s)=1 then s:='0'+s;
if pos(comma,s)=length(s) then delete(s,pos(comma,s),1);
if ms=true then s:='-'+s; end;
op_minus:=s; end;
//сложение со знаком
function plus(s1,s2:string):string;
begin
if (check(s1)=false) or (check(s2)=false) then begin
plus:='Enter correct operands!'; exit; end;
if (s1[1]='-') and (s2[1]='-') then begin
delete(s1,1,1); delete(s2,1,1);
plus:=op_plus(s1,s2); plus:='-'+plus; end else
if (s1[1]='-') and (s2[1]<>'-') then begin
delete(s1,1,1); plus:=op_minus(s2,s1); end else
if (s1[1]<>'-') and (s2[1]='-') then begin
delete(s2,1,1); plus:=op_minus(s1,s2) end
else plus:=op_plus(s1,s2); end;
//вычитание со знаком
function minus(s1,s2:string):string;
begin
if (check(s1)=false) or (check(s2)=false) then begin
minus:='Enter correct operands!'; exit; end;
minus:="";
if (s1[1]='-') and (s2[1]='-') then begin
delete(s1,1,1); delete(s2,1,1);
minus:=op_minus(s2,s1); end else
if (s1[1]='-') and (s2[1]<>'-') then begin
delete(s1,1,1); minus:=op_plus(s1,s2);
minus:='-'+minus; end else
if (s1[1]<>'-') and (s2[1]='-') then begin
delete(s2,1,1); minus:=op_plus(s1,s2) end
else minus:=op_minus(s1,s2); end;
//умножение
function mult(s1,s2:string):string;
var e,p,p2:string;
    i,j,l1,i1,j1,a,b,r,c,d,n,n1:longint;
    ms:boolean;
begin
if (check(s1)=false) or (check(s2)=false) then begin
mult:='Enter correct operands!'; exit; end;
op_plusb:=true; n1:=0; z1:=0; z2:=0; ms:=false;
if (s1[1]='-') or (s2[1]='-') then ms:=sign(s1,s2);
if (pos(comma,s1)>0) or (pos(comma,s2)>0) then begin
if pos(comma,s1)>0 then z1:=length(s1)-pos(comma,s1);
if pos(comma,s2)>0 then z2:=length(s2)-pos(comma,s2);
n1:=z1+z2;
if pos(comma,s1)>0 then delete(s1,pos(comma,s1),1);
if pos(comma,s2)>0 then delete(s2,pos(comma,s2),1);
if s1[1]='0' then s1:=zerodelb(s1);
if s2[1]='0' then s2:=zerodelb(s2); end;

```

```

if length(s1)<>length(s2) then eqsize(s1,s2);
n:=length(s1); r:=0; i:=0;
l1:=length(s2); p2:='0'; i1:=0;
while l1<>0 do begin
b:=chton(s2[l1]); dec(l1);
n:=length(s1); p:='0'; i:=0;
while n>0 do begin
a:=chton(s1[n]); dec(n);
r:=a*b; c:=r div sys; d:=r-(c*sys);
e:=ntoch(c)+ntoch(d); inc(i);
if i>0 then
for j:=1 to i-1 do e:=e+'0'; p:=op_plus(p,e);
end; inc(i1); if i1>0 then for j1:=1 to i1-1 do
p:=p+'0'; p2:=op_plus(p2,p); end;
if (n1>=length(p2)) then for i:=0 to n1-length(p2) do p2:='0'+p2;
if n1>0 then insert(comma,p2,length(p2)-n1+1);
if (pos(comma,p2)>0) and (p2[length(p2)]='0') then p2:=zerodele(p2);
if pos(comma,p2)=length(p2) then delete(p2,length(p2),1);
if ms=true then p2:='-'+p2;
mult:=p2; end;
//деление без остатка
function dividewr(s1,s2:string):string;
var r:string;
begin r:='0';
while (compare(s1,s2)='>') or (compare(s1,s2)='=') do begin
s1:=op_minus(s1,s2); r:=op_plus(r,'1');
end; od:=s1; od:=s1; dividewr:=r;
if dividewr="" then dividewr:='0'; end;
//деление в столбик
function edivide(s1,s2:string):string;
var o,r,t:string; i:byte;
begin
if (s1='0') or (s2='0') then begin
edivide:='0'; exit; end; i:=1;
while (compare(o,s2)<>'>') and (compare(o,s2)<>'=')
and (i<length(s1)) do begin
o:=o+s1[i]; inc(i); end;
if length(s1)<>length(s2) then begin
i:=1; o:=s1[1];
while i<length(s1)+1 do begin
t:=dividewr(o,s2); r:=r+t; t:=mult(r[length(r)],s2);
o:=minus(o,t); if o='0' then o:="";
inc(i); o:=o+s1[i]; end; end else r:=dividewr(s1,s2);
if r[1]='0' then r:=zerodelb(r);
delete(o,length(o),1);
edivide:=(r); end;
//деление с остатком
function divide(s1,s2:string):string;
var x,r,dp:string; ds:boolean; i,pc:integer;
begin
if (check(s1)=false) or (check(s2)=false) then begin
divide:='Enter correct operands!'; exit; end;

```

```

z1:=0; z2:=0; op_minusb:=true; ds:=false;
if (s1[1]='-') or (s2[1]='-') then
ds:=sign(s1,s2);
if (pos(comma,s1)>0) or (pos(comma,s2)>0) then begin
if pos(comma,s1)>0 then z1:=length(s1)-pos(comma,s1);
if pos(comma,s2)>0 then z2:=length(s2)-pos(comma,s2);
if z1>z2 then pc:=z1 else pc:=z2; z1:=0; z2:=0;
dp:='1'; for i:=1 to pc do dp:=dp+'0';
s1:=mult(s1,dp); s2:=mult(s2,dp); end;
x:=edivide(s1,s2); x:=x+comma; od:=od+'0';
if od<>'0' then
while (prec>0) and (od<>'00') do begin
r:={dividewr}edivide(od,s2); x:=x+r;
od:=od+'0'; dec(prec); end; od:=";
if pos(comma,x)=length(x) then delete(x,pos(comma,x),1);
if ds=true then x:='-'+x;
if pos(comma,x)=0 then divide:=x else divide:=zerodele(x); end;
//ВОЗВЕДЕНИЕ В СТЕПЕНЬ
function power(s,p:string):string;
var i:byte; t:string; b:boolean;
begin b:=false;
if p[1]='-' then begin delete(p,1,1); b:=true; end;
t:='1'; for i:=1 to strtoint(p) do t:=mult(s,t);
if b=true then t:=divide('1',t); power:=t; end;
//КОНВЕРТИРОВАТЬ ИЗ ДЕСЯТИЧНОЙ СИСТЕМЫ В ЛЮБУЮ ДРУГУЮ
function convertfromdec(s:string):string;
var r,fs,t,r1,fp,ip:string;
b:boolean; t1,pt:integer;
begin sys:=10; pt:=prec; b:=false; od:='0'; r:="; t:="; t1:=0; fs:=";
if pos(comma,s)=0 then fs:=s else if pos(comma,s)>0 then begin
b:=true; fs:=copy(s,1,pos(comma,s)-1);
delete(s,1,length(fs)); s:='0'+s; end;
if fs='0' then r:='0' else
while (prec>0) and (fs<>'0') do begin
fs:=edivide(fs,inttostr(sys2));
if od="" then od:='0';
t1:=strtoint(od); r:=ntoch(t1)+r; dec(prec); end;
r1:=r; r:="; if b=true then begin prec:=pt;
while (prec>0) and (fp<>'0') do begin
t:=copy(s,1,pos(comma,s)-1);
s:=mult(s,inttostr(sys2));
if pos(comma,s)=0 then ip:=s else ip:=copy(s,1,pos(comma,s)-1);
pt:=strtoint(ip); ip:=ntoch(pt); r:=r+ip;
if pos(comma,s)=0 then fp:='0';
if ip<>t then begin delete(s,1,pos(comma,s)-1); s:='0'+s; end;
dec(prec); end; end;
if r<>" then r1:=r1+comma+r;
if r1[1]='0' then delete(r1,1,1);
//if r1[1]=" then r1:='0';
if (r1[1]='0') and (pos(comma,r1)>2) then
while r1[1]='0' do delete(r1,1,1);
if r1[1]=comma then r1:='0'+r1;

```

```

convertfromdec:=r1; end;
//конвертер систем счисления
function numberconvert(s:string):string;
var i,k,n,pr,nst:integer; c:char; r1,ds,t1:string;
begin sys:=sys1; nst:=sys1; b:=false;
if s[1]='0' then delete(s,1,1);
if sys1=10 then begin
numberconvert:=convertfromdec(s); exit; end;
if (s[1]='0') and (pos(comma,s)>0) then begin
b:=true; delete(s,pos(comma,s)-1,1);
insert('1',s,0); end;
r1:='1'; sys:=10; ds:=""; b:=true;
if pos(comma,s)>0 then begin
ds:=copy(s,pos(comma,s)+1,length(s)-pos(comma,s));
delete(s,pos(comma,s),length(s)-pos(comma,s)+1); end;
k:= length(s)-1; i:=1; t1:='0';
while k>-1 do begin
r1:=power(inttostr(sys1),inttostr(k));
c:=s[length(s)-k]; n:=chton(c);
r1:=mult(inttostr(n),r1);
t1:=plus(r1,t1); dec(k); end;
if (ds<>") then begin
k:=-length(ds); i:=0;
for i:=1 to length(ds) do begin pr:=prec;
r1:=power(inttostr(sys1),inttostr(-i));
c:=ds[i]; n:=chton(c); prec:=pr;
r1:=mult(inttostr(n),r1); op_plusb:=false;
t1:=plus(r1,t1); end; end;
//if b=true then begin delete(t1,pos(comma,t1)-1,1);
//insert('0',t1,0); end;
if sys2<>10 then begin sys1:=10; t1:=convertfromdec(t1); end;
if t1[1]='0' then delete(t1,1,1);
if (t1[1]='0') and (pos(comma,t1)>2) then
while t1[1]='0' do delete(t1,1,1);
if t1[1]=comma then t1:='0'+t1;
numberconvert:=t1; sys1:=nst; end;
//обработка выражений без скобок
function strprocwb({var }s:string):string;
var s1,s2:string; c:char; pr,i:byte;
myset:set of char; b:boolean;
begin b:=false; pr:=prec; s1:="";
for i:=1 to length(s) do begin
if s[i]='(' then delete(s,i,1);
if s[i]=')' then delete(s,i,1); end;
set1:=set1+[comma];
for i:=48 to 48+sys-1 do
set1:=set1+[chr(i)];
if sys>9 then
for i:=65 to 65+sys-10-1 do
set1:=set1+[chr(i)]; i:=1;
if s[1]='-' then begin delete(s,1,1); b:=true; end;
while s[i] in set1 do begin s1:=s1+s[i]; inc(i); end;

```



```

delete(s,1,length(s1));
if b=true then s1:='-'+s1;
while s<>" do begin b:=false; i:=1;
prec:=pr; op_plusb:=false; op_minusb:=false; s2:=";
c:=s[1]; delete(s,1,1);
if s[1]='-' then begin delete(s,1,1); b:=true; end;
while s[i] in set1 do begin s2:=s2+s[i]; inc(i); end;
delete(s,1,length(s2));
if b=true then s2:='-'+s2;
case c of
'+': s1:=plus(s1,s2); '-': s1:=minus(s1,s2);
'*': s1:=mult(s1,s2); '/': s1:=divide(s1,s2); end;
end; strprocwb:=s1; end;
//обработка выражений со скобками
function strproc(s:string):string;
var r,t1,t2:string; p,i:byte;
begin
while s<>" do begin t2:="; i:=1;
if pos('(',s)>0 then begin
p:=pos('(',s); t1:=copy(s,1,p-1); delete(s,1,p);
p:=pos(')',s); t2:=copy(s,1,p-1); delete(s,1,p);
for i:=1 to length(t2) do begin
if t2[i]='(' then delete(t2,i,1);
if t2[i]=')' then delete(t2,i,1); end;
t2:=strprocwb(t2); end else begin t1:=s; s:="; end;
r:=r+t1+t2; end;
strproc:=strprocwb(r); end;
//запуск конвертера
function strprocconv(s:string):string;
var i,pr:byte; c,t1,r:string;
begin pr:=prec;
set1:=set1+[comma];
for i:=48 to 57 do
set1:=set1+[chr(i)];
for i:=65 to 90 do
set1:=set1+[chr(i)];
if not(s[length(s)] in set1) then begin
c:=s[length(s)]; delete(s,length(s),1); end;
while s<>" do begin i:=1; s2:="; t1:=";
op_plusb:=false; op_minusb:=false; prec:=pr;
while not(s[i] in set1) do begin
t1:=t1+s[i]; inc(i); end;
delete(s,1,length(t1)); i:=1; r:=r+t1;
while s[i] in set1 do begin
s2:=s2+s[i]; inc(i); end;
delete(s,1,length(s2));
s2:=numberconvert(s2); r:=r+s2; end;
r:=r+c; strprocconv:=r; end;
//=====КНОПКИ ВВОДА=====
procedure TForm1.charinput(ch:string);
begin
if b_ae=true then begin ae.text:=ae.text+ch;

```

```

ae.SetFocus; ae.SelStart:=length(ae.text); end else
if b_quantity=true then begin quantity.text:=quantity.text+ch;
quantity.SetFocus; quantity.SelStart:=length(quantity.text); end else
if b_thickness=true then begin thickness.text:=thickness.text+ch;
thickness.SetFocus; thickness.SelStart:=length(thickness.text); end else
if b_kgm=true then begin kgm.text:=kgm.text+ch;
kgm.SetFocus; kgm.SelStart:=length(kgm.text); end else
if b_rope_effort=true then begin rope_effort.text:=rope_effort.text+ch;
rope_effort.SetFocus; rope_effort.SelStart:=length(rope_effort.text); end else
if b_divprec=true then begin divprec.text:=divprec.text+ch;
divprec.SetFocus; divprec.SelStart:=length(divprec.text); end; end;
procedure TForm1.NumberButton(Sender: TObject);
var ch:string;
begin ch:=(Sender as TButton).hint; charinput(ch); end;
procedure TForm1.absClick(Sender: TObject);
begin if b_ae=true then begin ae.text:=ae.text+'-';
ae.SetFocus; ae.SelStart:=length(ae.text); end; end;
procedure TForm1.bracket1Click(Sender: TObject);
begin if b_ae=true then begin ae.text:=ae.text+'(';
ae.SetFocus; ae.SelStart:=length(ae.text); end; end;
procedure TForm1.bracket2Click(Sender: TObject);
begin if b_ae=true then begin ae.text:=ae.text+')';
ae.SetFocus; ae.SelStart:=length(ae.text); end; end;
procedure TForm1.b_plusClick(Sender: TObject);
begin if b_ae=true then begin ae.text:=ae.text+'+';
ae.SetFocus; ae.SelStart:=length(ae.text); end; end;
procedure TForm1.b_minusClick(Sender: TObject);
begin if b_ae=true then begin ae.text:=ae.text+'-';
ae.SetFocus; ae.SelStart:=length(ae.text); end; end;
procedure TForm1.b_multClick(Sender: TObject);
begin if b_ae=true then begin ae.text:=ae.text+'*';
ae.SetFocus; ae.SelStart:=length(ae.text); end; end;
procedure TForm1.b_divideClick(Sender: TObject);
begin if b_ae=true then begin ae.text:=ae.text+'/';
ae.SetFocus; ae.SelStart:=length(ae.text); end; end;
procedure TForm1.buttonenable(n:byte);
var i:byte; a:array[0..35] of boolean;
begin
for i:=0 to 35 do a[i]:=false;
for i:=0 to n-1 do a[i]:=true;
if a[0]=true then b0.Enabled:=true else b0.Enabled:=false;
if a[1]=true then b1.Enabled:=true else b1.Enabled:=false;
if a[2]=true then b2.Enabled:=true else b2.Enabled:=false;
if a[3]=true then b3.Enabled:=true else b3.Enabled:=false;
if a[4]=true then b4.Enabled:=true else b4.Enabled:=false;
if a[5]=true then b5.Enabled:=true else b5.Enabled:=false;
if a[6]=true then b6.Enabled:=true else b6.Enabled:=false;
if a[7]=true then b7.Enabled:=true else b7.Enabled:=false;
if a[8]=true then b8.Enabled:=true else b8.Enabled:=false;
if a[9]=true then b9.Enabled:=true else b9.Enabled:=false;
if a[10]=true then b10.Enabled:=true else b10.Enabled:=false;
if a[11]=true then b11.Enabled:=true else b11.Enabled:=false;

```

```

if a[12]=true then b12.Enabled:=true else b12.Enabled:=false;
if a[13]=true then b13.Enabled:=true else b13.Enabled:=false;
if a[14]=true then b14.Enabled:=true else b14.Enabled:=false;
if a[15]=true then b15.Enabled:=true else b15.Enabled:=false;
if a[16]=true then b16.Enabled:=true else b16.Enabled:=false;
if a[17]=true then b17.Enabled:=true else b17.Enabled:=false;
if a[18]=true then b18.Enabled:=true else b18.Enabled:=false;
if a[19]=true then b19.Enabled:=true else b19.Enabled:=false;
if a[20]=true then b20.Enabled:=true else b20.Enabled:=false;
if a[21]=true then b21.Enabled:=true else b21.Enabled:=false;
if a[22]=true then b22.Enabled:=true else b22.Enabled:=false;
if a[23]=true then b23.Enabled:=true else b23.Enabled:=false;
if a[24]=true then b24.Enabled:=true else b24.Enabled:=false;
if a[25]=true then b25.Enabled:=true else b25.Enabled:=false;
if a[26]=true then b26.Enabled:=true else b26.Enabled:=false;
if a[27]=true then b27.Enabled:=true else b27.Enabled:=false;
if a[28]=true then b28.Enabled:=true else b28.Enabled:=false;
if a[29]=true then b29.Enabled:=true else b29.Enabled:=false;
if a[30]=true then b30.Enabled:=true else b30.Enabled:=false;
if a[31]=true then b31.Enabled:=true else b31.Enabled:=false;
if a[32]=true then b32.Enabled:=true else b32.Enabled:=false;
if a[33]=true then b33.Enabled:=true else b33.Enabled:=false;
if a[34]=true then b34.Enabled:=true else b34.Enabled:=false;
if a[35]=true then b35.Enabled:=true else b35.Enabled:=false; end;

```

//=====калькулятор и

конвертер=====

```

procedure TForm1.nsChange(Sender: TObject);
var n:byte;
begin n:=strtoint(ns.text);
buttonenable(n); end;
procedure TForm1.ns2Click(Sender: TObject);
begin ns.text:='2'; buttonenable(2); end;
procedure TForm1.ns8Click(Sender: TObject);
begin ns.text:='8'; buttonenable(8); end;
procedure TForm1.ns10Click(Sender: TObject);
begin ns.text:='10'; buttonenable(10); end;
procedure TForm1.ns16Click(Sender: TObject);
begin ns.text:='16'; buttonenable(16); end;
procedure TForm1.divprecClick(Sender: TObject);
begin
  b_ae:=false; b_quantity:=false;
  b_thickness:=false; b_kgm:=false;
  b_rope_effort:=false; b_divprec:=true;
end;
procedure TForm1.aeClick(Sender: TObject);
begin
  b_ae:=true; b_quantity:=false;
  b_thickness:=false; b_kgm:=false;
  b_rope_effort:=false; b_divprec:=false;
end;
procedure TForm1.TestModeChange(Sender: TObject);
begin

```

```

if TestMode.State=cbUnchecked then begin
bracket1.enabled:=false; bracket2.enabled:=false; end
else begin
bracket1.enabled:=true; bracket2.enabled:=true; end; end;
procedure TForm1.Convop1Click(Sender: TObject);
var s:string; t : Longword; tr:real;
begin converter:=true;
op_plusb:=false; op_minusb:=false; prec:=strtoint(divprec.text);
sys1:=strtoint(system1.text); sys2:=strtoint(system2.text);
s:=ae.text;
t:=GetTickCount; e3.text:=numberconvert(s);//strprocconv(s);
t:=GetTickCount-t;
tr:=t; tr:= tr/1000;
memo.text:=memo.text+s+' in system '+system1.text+
' = '+e3.text+' in system '+system2.text;//+';'+#13; end;
if calctime.state=cbunChecked then memo.text:=memo.text+';'+#13 else
memo.text:=memo.text+' Время расчета: '+
FormatFloat('0.####',tr)+' секунд '+#13+#13; ae.SetFocus; end;
procedure TForm1.clear_opClick(Sender: TObject);
begin ae.text:="";
e3.text:=""; ae.SetFocus; end;
procedure TForm1.clear_jClick(Sender: TObject);
begin memo.Text:=""; ae.SetFocus; end;
procedure TForm1.ClearallClick(Sender: TObject);
begin e3.text:=""; ae.SetFocus; end;
procedure TForm1.resClick(Sender: TObject);
var s:string; t : Longword; tr:real;
begin converter:=false;
if ae.text<>" then begin
op_plusb:=false; op_minusb:=false; s:=ae.text;
sys:=strtoint(ns.text); prec:=strtoint(divprec.text);
t:=GetTickCount; e3.text:=strproc(s); t:=GetTickCount-t;
tr:=t; tr:= tr/1000;
memo.text:=memo.text+ae.text+'='+e3.text+' ('+ns.text+')';
if calctime.state=cbunChecked then memo.text:=memo.text+';'+#13 else
memo.text:=memo.text+' Время расчета: '+
FormatFloat('0.####',tr)+' секунд '+#13+#13;
end else showmessage('Отсутствует операция!');
ae.SetFocus; end;
procedure TForm1.clearall_materialClick(Sender: TObject);
begin
quantity.Text:=""; thickness.text:="";
res_weight.text:="";
kgm.text:=""; weight_or_length.text:="";
rope_effort.Text:="";
bursting_effort.text:=""; rope_d.text:="";
ae.Text:=""; e3.text:=""; memo.text:="";
ae.SetFocus; end;
procedure TForm1.FormKeyDown(Sender: TObject; var Key: Word);
begin if (Key=13) then resClick(Sender); end;
procedure TForm1.system1Change(Sender: TObject);
var n:byte;

```

```

begin n:=strtoint(system1.text);
buttonenable(n); end;
procedure TForm1.MemoChange(Sender: TObject);
begin
    Memo.SelStart:=Length(Memo.Text);
    Memo.SelLength:=0;
    sendmessage(Memo.Handle, EM_SCROLLCARET, 0, 0); end;
//=====Металлопрокат=====
function checkfield(s:string):boolean;
var set1:set of char; i:byte;
begin
    if s="" then begin checkfield:= false;
    showmessage('Введите корректные данные!'); exit; end; set1:=[];
    for i:=48 to 57 do set1:=set1+[chr(i)]; set1:=set1+[comma];
    for i:=1 to length(s) do
        if not(s[i] in set1) then begin
            checkfield:=false; showmessage('Введите корректные данные!'); exit; end;
            checkfield:=true; end;
procedure TForm1.MetallTypeChange(Sender: TObject);
begin
    if (MetallType.text='Сопровой') or (MetallType.text='Трубы')
    then begin res_weight.Text:="";
    thickness.Enabled:=false;
    thickness.text:="";
    GostMetallType.Enabled:=true;
    unit_weight.Enabled:=true;
    if MetallType.text='Сопровой' then begin
        GostMetallType.Items.LoadFromFile('Names&Dimensions_sorted.txt');
        unit_weight.Text:='0,038';
    end else begin
        GostMetallType.Items.LoadFromFile('Names&Dimensions_pipe.txt');
        unit_weight.Text:='0,84';
    end;
    GostMetallType.ItemIndex:=0;
end
else begin res_weight.Text:="";
    thickness.Enabled:=true;
    GostMetallType.Enabled:=false;
    GostMetallType.Items.clear;
    GostMetallType.Text:="";
    unit_weight.Enabled:=false;
    unit_weight.Text:=""; end; end;
procedure TForm1.quantityClick(Sender: TObject);
begin
    b_ae:=false; b_quantity:=true;
    b_thickness:=false; b_kgm:=false;
    b_rope_effort:=false; b_divprec:=false;
end;
procedure TForm1.thicknessClick(Sender: TObject);
begin
    b_ae:=false; b_quantity:=false;
    b_thickness:=true; b_kgm:=false;

```

```

    b_rope_effort:=false; b_divprec:=false;
end;
procedure TForm1.GostMetallTypeChange(Sender: TObject);
var f:text; s:string; i:integer;
begin
    if MetallType.Text='Сортовой' then
        assignfile(f,'Weights_sorted.txt')
    else assignfile(f,'Weights_pipe.txt');
    reset(f);
    for i:=0 to GostMetallType.ItemIndex do
        readln(f,s);
        unit_weight.Text:=s;
    closefile(f); end;
    procedure TForm1.CalcWeightClick(Sender: TObject);
    var s:string;
    begin s:=""; ns.text:='10';
    if MetallType.Text='Листовой' then begin
        if (checkfield(quantity.text)=true) and (checkfield(thickness.text)=true) then
            s:=s+quantity.text+'*'+thickness.text+'*'+7,85' end else
            if (MetallType.Text='Сортовой') or (MetallType.Text='Трубы') then
                if checkfield(quantity.text)=true then
                    s:=s+quantity.text+'*'+unit_weight.text;
                if s="" then quantity.SetFocus else begin
                    ae.text:=s; resClick(Sender);
                    res_weight.text:=e3.text; end; quantity.SetFocus; end;
                //=====Вес и длина троса=====
    procedure TForm1.type_of_sizeChange(Sender: TObject);
    var f:text; s:string;
    begin
        assignfile(f,'rope_size&weight.txt');
        reset(f); rope_s.Items.clear;
        if type_of_size.Text='по окружности' then
            while not eof(f) do begin
                readln(f,s);
                delete(s,pos(' ',s),length(s)-pos(' ',s)+1);
                rope_s.Items.Add(s);
            end else
                while not eof(f) do begin s:="";
                readln(f,s);
                delete(s,1,pos(' ',s));
                delete(s,pos('/',s),length(s)-pos('/',s)+1);
                rope_s.Items.Add(s);
            end;
            closefile(f); rope_s.itemindex:=0;
            rope_weight.Text:='0.068'; end;
    procedure TForm1.rope_sChange(Sender: TObject);
    var i:integer; f:text;
    begin
        assignfile(f,'rope_size&weight.txt');
        reset(f);
        for i:=0 to rope_s.ItemIndex do
            readln(f,s);

```

```

delete(s,1,pos('/',s));
rope_weight.text:=s;
end;
procedure TForm1.kgmClick(Sender: TObject);
begin
  b_ae:=false; b_quantity:=false;
  b_thickness:=false; b_kgm:=true;
  b_rope_effort:=false; b_divprec:=false;
end;
procedure TForm1.calc_w_lClick(Sender: TObject);
begin
  if checkfield(kgm.text)=false then begin kgm.SetFocus; exit; end;
  ns.text:='10'; ae.text:=kgm.text;
  if unit_w_l.text='век' then ae.Text:=ae.text+'*'
  else ae.Text:=ae.text+'/';
  ae.Text:=ae.text+rope_weight.text;
  resClick(Sender); weight_or_length.text:=e3.text;
  if unit_w_l.text='век' then
    weight_or_length.text:=weight_or_length.text+' кг'
  else weight_or_length.text:=weight_or_length.text+' м';
  kgm.SetFocus; end;
procedure TForm1.unit_w_lChange(Sender: TObject);
begin
  if unit_w_l.text='век' then begin
    calc_w_l.Caption:='Вычислить вес троса';
    label11.Caption:='Вес троса (кг)';
    m_unit1.caption:='Количество метров'; end else begin
    calc_w_l.Caption:='Вычислить длину троса';
    m_unit1.caption:='Количество килограмм';
    label11.Caption:='Длина троса (м)'; end;
end;
//=====Разрывное усилие и диаметр
каната=====
procedure TForm1.rope_condChange(Sender: TObject);
var f:text; i:integer; s:string;
begin
  assignfile(f,'rope-cond.txt');
  reset(f);
  for i:=0 to rope_cond.ItemIndex do
    readln(f,s);
    delete(s,1,pos('/',s));
    coefficient.text:=s; end;
procedure TForm1.rope_typeChange(Sender: TObject);
var f:text; s:string;
begin
  if bursting_effort.text='' then exit;
  if rope_type.text='трехрядный пеньковый' then
    assignfile(f, 'rope-diameter_hemp.txt') else
  if rope_type.text='стальной (6х36 - 216 проволок)' then
    assignfile(f, 'rope-diameter_iron216.txt') else
    assignfile(f, 'rope-diameter_iron222.txt'); reset(f);
  while not (eof(f)) do begin

```

```

readln(f,s);
if strtoint(bursting_effort.text)<=strtoint(copy(s,1,pos('/',s)-1))
then begin delete(s,1,pos('/',s));
rope_d.text:=s; closefile(f); exit; end; end;
rope_d.text:='Большое усилие!'; closefile(f);
end;
procedure TForm1.rope_effortClick(Sender: TObject);
begin
  b_ae:=false; b_quantity:=false;
  b_thickness:=false; b_kgm:=false;
  b_rope_effort:=true; b_divprec:=false;
end;
procedure TForm1.calc_effortClick(Sender: TObject);
var f:text; s:string;
begin
  if checkfield(rope_effort.text)=false then begin
    rope_effort.SetFocus; exit; end; ns.text:='10';
    ae.text:=rope_effort.text+'*'+coefficient.text;
    resClick(Sender);
    if pos(comma,e3.text)=0 then bursting_effort.text:=e3.text else
    bursting_effort.text:=inttostr(round(strtoint(e3.text)));
    if rope_type.text='трехрядный пеньковый' then
    assignfile(f, 'rope-diameter_hemp.txt') else
    if rope_type.text='стальной (6х36 - 216 проволоки)' then
    assignfile(f, 'rope-diameter_iron216.txt') else
    assignfile(f, 'rope-diameter_iron222.txt'); reset(f);
    while not (eof(f)) do begin
      readln(f,s);
      if strtoint(bursting_effort.text)<=strtoint(copy(s,1,pos('/',s)-1))
      then begin delete(s,1,pos('/',s));
rope_d.text:=s; closefile(f); exit; end; end;
rope_d.text:='Большое усилие!'; closefile(f);
rope_effort.SetFocus; end;
//=====
procedure TForm1.deactivate_screenkeyboard(Sender: TObject);
begin
  b_ae:=false; b_quantity:=false;
  b_thickness:=false; b_kgm:=false;
  b_rope_effort:=false; b_divprec:=false;
end;
procedure TForm1.FormCreate(Sender: TObject);
begin buttonenable(10); aeClick(Sender); comma:=','; end;
end.

```