

unit unit_psw1;

{ \$mode objfpc } { \$H+ }

interface

uses

Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, StdCtrls; ↗

type

{ TForm1 }

TForm1 = **class**(TForm)

Button1: TButton; {запоминание заданного пароля}

Button2: TButton; {запуск процедуры подбора}

Edit1: TEdit; {поле ввода пароля, который программа подбора
будет угадывать} ↗

Label1: TLabel; {поясняет назначение приложения}

Label2: TLabel; {Инструкция пользователю}

Label3: TLabel; {Вывод числа попыток до успеха}

Label4: TLabel; {Подобранный пароль}

procedure Button1Click(Sender: TObject);

procedure Button2Click(Sender: TObject);

private

{ private declarations }

public

{ public declarations }

end;

const

L3= 'число попыток подбора ';

L4= 'Подобранный пароль ';

L4a= '*****';

var

Form1: TForm1;

en : boolean; {Флаг принятия неприятия заданного пароля}

password: **string**[6]; {Образец пароля, хранящийся в программе.}

implementation

{ \$R *.lfm }

{Процедура, ПОДБИРАЮЩАЯ ПАРОЛЬ, прямым перебором}

procedure try_passw(**var** N: Int64; **var** S: **string**);

{Пробуем строить его из предыдущего пароля,
перебирая коды ASCII таблицы из разрешённых диапазонов}

const

min1=65;

max1=90;

min2=97;

max2=122;

var

ctnum : **byte**;

maxnum : **byte**;

{CtChar : Char;}

CodChar: **byte**;

p : **byte**; {Флаг "переноса"}

begin

{Для 1-го вызова возвращаем начальную комбинацию}

{Если длинна пароля переменная, надо усложнить генерацию начальной комбинации} ↵

if N=1 **then** S:='AAAAAA';

N:=N+1; {Увеличиваем счёт вызовов}

{Определяем максимальный номер элемента в строке}

maxnum:= Length(S);

{Заводим перебор}

ctnum:=maxnum;

Repeat

Codchar:=ord(S[ctnum]);

p:=0;

{Изменяем пароль только, если он состоит из разрешенных символов }

if ((CodChar≥min1)**and**(CodChar≤max1))**or**((CodChar≥min2)**and**(CodChar≤max2)) ↵

then

begin

{получаем следующую комбинацию}

if ((CodChar≥min1)**and**(CodChar<max1))

then CodChar:=CodChar+1

else if (CodChar=max1)

then CodChar:=min2

else if ((CodChar≥min2)**and**(CodChar<max2))

then CodChar:=CodChar+1

```
                else if (CodChar=max2)
                    then
                        begin
                            CodChar:=min1;
                            p:=1
                        end;
                S[ctnum]:=Chr(CodChar);
            end;

        if p=1 then
            begin
                ctnum:=ctnum-1;
                {Зацикливаем перебор. Возвращаем начальную комбинацию}
                if ctnum<1 then S:='AAAAAA';
            end;
        Until p=0;
    end;

{ TForm1 }

{Приёмка образца}
{Процедура, проверяющая заданный ПАРОЛЬ, проверяем латинские буквы и
усекаем до 6}
procedure TForm1.Button1Click(Sender: TObject);
const
    MS1='Образец пароля принят';
    MS2='Образец не принят. Должно быть 6 латинских букв!';

    var i : integer; {счётчик}
        S : string[6]; {образ заданного пароля}

begin
    en:=true;      {Надеемся, что образец пароля будет принят}
    S:=Edit1.text; {Считываем образец пароля из компонента Edit1}

    {Проверки}
    {Если длина меньше 6 символов, не принимаем}
    if Length(S)<6 then en:=false;
    {Если символы в строке не попадают в заданный набор латинских букв,
    не принимаем}
    For i:=1 to 6 DO
        begin
            if (ord(S[i])<65)or((ord(S[i])>90)and(ord(S[i])<97))or(ord(S[i]
            ])>122)
```

```
    then
        begin
            en:=false;
            break;
        end;
    end;

{Выводим соответствующие сообщения}
if en
then
    begin
        {Выводим в поле Edit1.Text урезанный до 6 символов пароль}
        Edit1.Text:= Copy(S,1,6);
        {Сохраним образец}
        password:= Copy(S,1,6);

        {Вывод в метки Label3, Label4}
        Label3.Caption:=L3+'?';
        Label4.Caption:=L4+L4a;

        {Сообщение о принятии образца}
        MessageDlg(MS1, mtInformation, [mbOk],0);
    end
else MessageDlg(MS2, mtError, [mbOk],0);

end;

procedure TForm1.Button2Click(Sender: TObject);
const
    MS1='Нет принятого образца';
var
    k          : Int64 ; {Счетчик попыток}
    m          : Integer;
    PSW        : String;{ [6];} {Проверяемый образец пароля}
begin
    {Если есть принятый образец пароля, начинаем подбирать}
    if en
    then
        begin
            k:=1;
            m:=1;
            Repeat
                try_passw(k, PSW);
```

```
        {m:=m+1;}
        {Вывод промежуточных результатов в метки Label3, Label4}
        {Хотя он всё равно не работает, комментируем}
        {If (m≥1000) then
            begin
                Label3.Caption:=L3+intToStr(k);
                Label4.Caption:=L4+PSW;
                m:=0;
            end;
        }
    until (PSW=password);

    {Вывод окончательных результатов в метки Label3, Label4}
    Label3.Caption:=L3+intToStr(k);
    Label4.Caption:=L4+PSW;
end

{Если принятого образца нет, выводим сообщение}
else MessageDlg(MS1, mtError, [mbOk],0);
end;

end.
```