```
PDEParabMp := proc (Listt, n, m, a, b, g, e, f, alfa1, beta1, alfa2, beta2, gama1, gama2, phi)
    local h, n1, u, i, j, h2, x, drov, d1, d2, d3, u0, uv, up, pp, p, q, r, s, pom, res;
    h := evalf((b-a)/n);
    n1 := n+1;
    h2 := h^2;
    x := [seq(a+(i-1)*h, i=1..n1)];
    u0 := [seq(phi(x[i]), i=1..n1)];
    uv := Matrix((m+1, n1), 0);
    d1 := [seq(0, i=1..n-1)]; print(op(d1));
    d2 := [seq(0, i=1..n-1)];
    d3 := [seq(0, i=1..n-1)];
    up := [seq(0, i=1..n-1)];
    pp := [seq(0, i=1..n-1)];
    drov := [seq(0, i=1..n-1)];
        #napojítání koeficientTv diferenciální rovnici
    for i from 1 to n-1 do
        d1[i] := evalf(g(x[i+1], j*k)/h2-e(x[i+1], j*k)/(2*h));
        d2[i] := evalf(-2*g(x[i+1], j*k)/h2);
        d3[i] := evalf(g(x[i+1], j*k)/h2+e(x[i+1], j*k)/(2*h));
    end do;
        # Dosezení okrajových podmínek do první a poslední rovnice
        p := 1/(2*h*alfa1-3*beta1(t));
        r := beta1(t)*p;
        pom := d1[1]*r;
        d2[1] := d2[1]-4*pom;
        d3[1] := d3[1]+pom;
        q := 1/(2*h*alfa2+3*beta2(t));
        s := beta2(t)*q;
        pom := d3[n-1]*s;
        d2[n-1] := d2[n-1]+4*pom;
        d1[n-1] := d1[n-1]-pom;
        # VýtvoŠení soustavy diferenciálních rovnic
        for i from 1 to n-1 do
            up[i] := u[i](t);
            pp[i] := u[i](0) = u0[i];
        end do;
        for i from 2 to n-2 do
            drov[i] := D(u[i])(t) = d1[i]*u[i-1](t)+d2[i]*u[i](t)+d3[i]*u[i+1](t)
    +f(x[i+1], t, u[i](t));
        end do;
        drov[1] := D(u[1])(t) = d1[1]*gama1(t)*2*h*p+d2[1]*u[1](t)+d3[1]
    *u[2](t)+f(x[2], t, u[1](t));
        drov[n-1] := D(u[n-1])(t) = d1[n-1]*u[n-2](t)+d2[n-1]*u[n-1](t)+d3[n-1]
    *gama2(t)*2*h*q+f(x[n], t, u[n-1](t));
        # výpoštet soustavy diferenciálních rovnic
        res := dsolve([op(drov), op(pp)], up, numeric);
        # Výpoštet tabulky hodnot Šešení
        for i from 1 to n1 do
            uv[1, i] := u0[i];
        end do;
        for j from 2 to m+1 do
            for i from 2 to n do
```

```
            uv[ j, i ] := rhs( ( res(Listt[ j ]) ) [ i ]);
        end do;
        uv[ j, 1 ] := gama1(Listt[ j ]) * 2 * h * p - 4 * r * uv[ j, 2 ] + r * uv[ j, 3 ];
        uv[ j, n1 ] := gama2(Listt[ j ]) * 2 * h * q + 4 * s * uv[ j, n ] - s * uv[ j, n - 1 ];
    end do;
  RETURN ( eval( uv ) );
end proc:
```

PŠíklad 1:

```
phi := x→sin( evalf ( Pi·x ) ) :
alfa1 := 1 :
 beta1 := t→0 :
alfa2 := 1 :
 beta2 := t→0 :
gama1 := t→0 :
 gama2 := t→0 :
g := ( x, t ) →1 :
 e := ( x, t ) →0 :
f := ( x, t, y ) →0 :
 n := 10 :
m := 10 :
h := 1.0 / n ;
 T := 1 ;
 Listt := [ seq( evalf ( i·T / m ), i = 0 ..m ) ];
```

$$0.1000000000$$

$$1$$

$$[0., 0.1000000000, 0.2000000000, 0.3000000000, 0.4000000000, 0.5000000000, 0.6000000000, 0.7000000000, 0.8000000000, 0.9000000000, 1.] \quad \textbf{(1.1)}$$

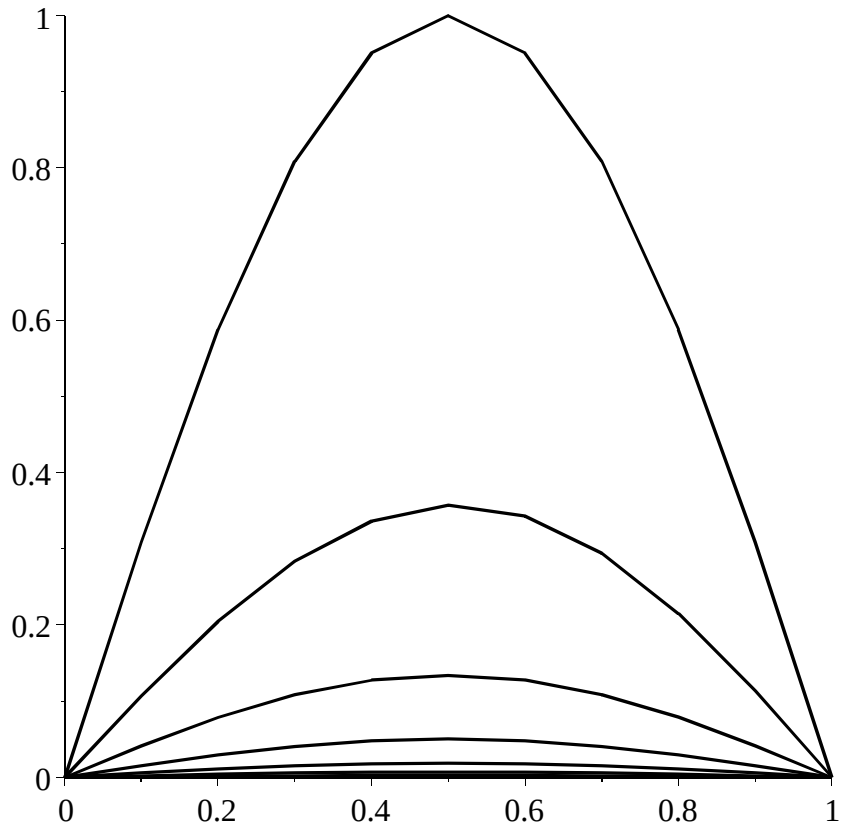
```
vys := PDEParabMp(Listt, n, m, 0.0, 1.0, g, e, f, alfa1, beta1, alfa2, beta2, gama1, gama2, phi);
```

$$0, 0, 0, 0, 0, 0, 0, 0, 0, 0$$

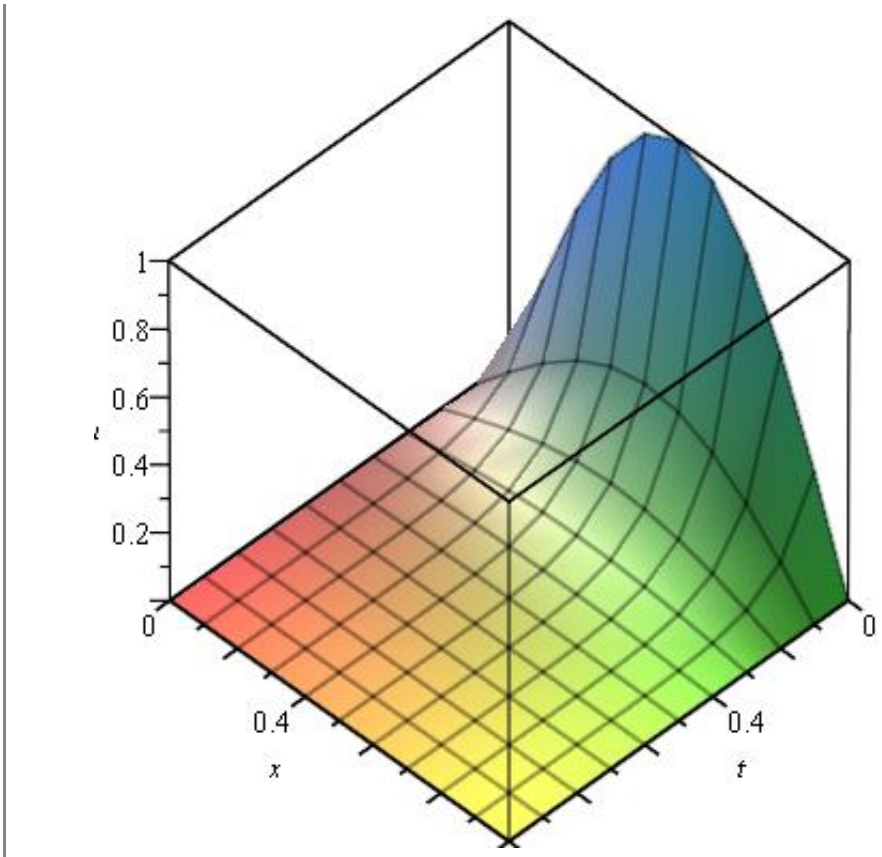
$$\begin{bmatrix} \textit{11 x 11 Matrix} \\ \textit{Data Type: anything} \\ \textit{Storage: rectangular} \\ \textit{Order: Fortran\_order} \end{bmatrix} \quad \textbf{(1.2)}$$

```
data := [ seq( [ seq( [ 0 + ( i − 1 ) ·h, vys[ j, i ] ], i = 1 ..n + 1 ) ], j = 1 .. m + 1 ) ] :
with( plots ) :


display( seq( listplot( data[ i ] ), i = 1 ..m ) );
```




```
a := subs(1 .. m + 1 = 0 ..T, 1 .. n + 1 = 0 .. 1, matrixplot(vys[1 .. m + 1, 1 .. n + 1], labels = [t,
     x, u])) :
display(a, view = [0 .. T, 0 .. 1, 0 .. 1]);
```


1
0.8
0.6
u
0.4
0.2
0
0.4
x
0
0.4
t