



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA STROJNÍHO INŽENÝRSTVÍ**  
**ENERGETICKÝ ÚSTAV**

FACULTY OF MECHANICAL ENGINEERING  
ENERGY INSTITUTE

**TERMODYNAMIKA IDEÁLNÍCH PLYNŮ V MATLABU**  
THERMODYNAMICS OF IDEAL GASES IN MATLAB

**BAKALÁŘSKÁ PRÁCE**  
BACHELOR'S THESIS

**AUTOR PRÁCE**  
AUTHOR

**JAKUB ZÁBOJNÍK**

**VEDOUCÍ PRÁCE**  
SUPERVISOR

**doc. Ing. JOSEF ŠTĚTINA, Ph.D.**

BRNO 2012



Vysoké učení technické v Brně, Fakulta strojního inženýrství

Energetický ústav

Akademický rok: 2011/2012

## **ZADÁNÍ BAKALÁŘSKÉ PRÁCE**

student(ka): Jakub Zábojník

který/která studuje v **bakalářském studijním programu**

obor: **Strojní inženýrství (2301R016)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma bakalářské práce:

### **Termodynamika ideálních plynů v Matlabu**

v anglickém jazyce:

### **Thermodynamics of ideal gases in Matlab**

Stručná charakteristika problematiky úkolu:

Cílem je vytvořit knihovnu funkcí (toolbox) pro základní termodynamické výpočty s ideálními plyny a se směsmi ideálních plynů. Knihovna by měla být doplněna i skripty pro vykreslování grafů dějů a oběhů.

Cíle bakalářské práce:

V prostředí MATLAB případně Octave vytvořte základní sadu funkcí (toolbox) pro řešení termodynamických výpočtů s ideálními plyny. K těmto funkcím zpracujte popis. Výpočty i popisy by měly být doplněny grafy p-v a T-s diagramy, proto vykreslování těchto grafů zpracujte opět v Matlabu.

Seznam odborné literatury:

- [1] Hahn Brian, Essential MATLAB, Elsevier 2010
- [2] Palm Wiliam, Introduction to MATLAB for Engineers, McGrawHill 2005
- [3] Massoud M., Engineering Thermofluids, Springer 2005
- [4] Balmer Robert, Modern Engineering Thermodynamics, Elsevier 2011

Vedoucí bakalářské práce: doc. Ing. Josef Štětina, Ph.D.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2011/2012.

V Brně, dne 19.11.2011

L.S.

---

doc. Ing. Zdeněk Skála, CSc.  
Ředitel ústavu

---

prof. RNDr. Miroslav Doupovec, CSc., dr. h. c.  
Děkan fakulty

## **ABSTRAKT**

V rámci bakalářské práce byla v prostředí programu MATLAB vytvořena sada funkcí pro řešení základních termodynamických výpočtů s ideálními plyny a se směsmi ideálních plynů. Do práce jsou zahrnuty vedle zdrojových textů těchto funkcí také jejich popisy. Dále jsou uvedeny základní charakteristiky termodynamických veličin a pojmů, na jejichž znalosti bylo při tvorbě funkcí stavěno, a návody k volání jednotlivých funkcí. Knihovna funkcí je realizována tak, že funkce jsou pomyslně rozděleny do několika úrovní (skupin), přičemž toto rozdělení vychází z principu termodynamického výpočtu a platí, že funkce ve dvou sousedních úrovních na sebe přímo navazují. Pro větší přehlednost návaznosti funkcí a pro přiblížení využití jednotlivých funkcí i knihovny samotné, zahrnuje práce také příklady použití při konkrétních termodynamických výpočtech.

## **ABSTRACT**

In terms of bachelor's thesis a toolbox of functions for basic computing in thermodynamics with ideal gases and their mixtures was built in environment of computer program MATLAB. Source code of these functions and their description are parts of this thesis. Among others fundamental characteristics of thermodynamic values and terms on which the creation of functions was built, and instruction for calling of functions are mentioned here. Functions in toolbox imaginary are divided to some groups. This dividing is results from principle of thermodynamic calculation. Functions from two neighbouring groups are directly tied together. For lucidity of function sequences and for functions usage and toolbox explanation the examples of using functions examples of functions usage in specific thermodynamic calculations.

## **KLÍČOVÁ SLOVA**

Termodynamika, MATLAB, knihovna funkcí, ideální plyn, směs ideálních plynů, termodynamický děj, termodynamický cyklus, stavová veličina

## **KEYWORDS**

Thermodynamics, MATLAB, toolbox, ideal gas, mixture of ideal gasses, thermodynamic process, thermodynamic cycle, state quantity

## **BIBLIOGRAFICKÁ CITACE**

ZÁBOJNÍK, J. *Termodynamika ideálních plynů v Matlabu*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2012. 82 s. Vedoucí bakalářské práce doc. Ing. Josef Štětina, Ph.D.

## **PROHLÁŠENÍ**

Prohlašuji, že jsem bakalářskou práci na téma Termodynamika ideálních plynů v MATLABu vypracoval samostatně s použitím odborné literatury a pramenů uvedených na seznamu, který tvoří přílohu této práce.

24. května 2012

.....

Jakub Zábojník

## **PODĚKOVÁNÍ**

Na tomto místě bych chtěl poděkovat především doc. Ing. Josefu Štětinovi, Ph.D. za cenné rady a připomínky poskytnuté při vypracování bakalářské práce. Poděkování patří také mé rodině za podporu poskytnutou během studia i při tvorbě této práce.

# OBSAH

<b>ÚVOD .....</b>	<b>11</b>
<b>1 MATLAB – ZÁKLADNÍ POJMY .....</b>	<b>12</b>
1.1 ÚVOD K PROGRAMU MATLAB	12
1.2 ZÁKLADNÍ POJMY PROGRAMOVACÍHO PROSTŘEDÍ MATLAB	12
1.3 FUNKCE MATLABU POUŽITÉ V BAKALÁŘSKÉ PRÁCI	13
<b>2 ZÁKLADNÍ FUNCE .....</b>	<b>16</b>
2.1 VOLBA PRACOVNÍ LÁTKY	16
2.2 VLASTNOSTI PLYNU	18
2.3 STAVOVÁ ROVNICE	21
2.4 PŘEVODY VELIČIN	22
2.4.1 PŘEVODY TLAKU Z Pa NA kPa A NAOPAK	22
2.4.2 PŘEVODY TEPLoty ZE °C NA K A NAOPAK	23
<b>3 TERMODYNAMICKÉ DĚJE.....</b>	<b>24</b>
3.1 ZÁKLADNÍ TEORIE TERMODYNAMICKÝCH DĚJŮ	24
3.2 NÁZVY FUNKCÍ	24
3.3 PRINCIP VÝPOČTŮ VE FUNKCÍCH	24
3.3.1 IZOBARICKÝ, IZOCHORICKÝ A IZOTERMICKÝ DĚJ	24
3.3.2 ADIABATICKÝ A POLYTROPICKÝ DĚJ	26
3.4 VOLÁNÍ FUNKCÍ	27
3.5 OŠETŘENÍ VSTUPŮ	27
<b>4 FUNKCE TERMODYNAMICKÝCH DĚJŮ.....</b>	<b>28</b>
4.1 IZOBARICKÝ DĚJ	28
4.2 IZOCHORICKÝ DĚJ	30
4.3 IZOTERMICKÝ DĚJ	32
4.4 ADIABATICKÝ DĚJ	34
4.5 POLYTROPICKÝ DĚJ	36
4.6 POMOCNÉ FUNKCE TERMODYNAMICKÝCH DĚJŮ	37
4.6.1 FUNKCE <i>APSTAV</i>	37
4.6.2 FUNKCE <i>STAVY</i> A <i>STAVYD</i>	39
<b>5 ENERGETICKÉ VELIČINY TERMODYNAMICKÝCH DĚJŮ .....</b>	<b>40</b>
5.1 OBJEMOVÁ PRÁCE	40
5.2 TECHNICKÁ PRÁCE	41
5.3 TEPLO	43
<b>6 VYKRESLENÍ GRAFŮ.....</b>	<b>45</b>
6.1 VYKRESLENÍ <i>p-V</i> DIAGRAMU	45
6.2 VYKRESLENÍ <i>T-S</i> DIAGRAMU	46
<b>7 APLIKACE KNIHOVNY FUNKCÍ .....</b>	<b>47</b>
7.1 VÝPOČET PŘÍKLADU S VYUŽITÍM KNIHOVNY FUNKCÍ	47
7.2 UŽITEČNÉ TIPY PRO UŽIVATELE	51

<b>ZÁVĚR.....</b>	<b>52</b>
<b>SEZNAM POUŽITÝCH ZDROJŮ.....</b>	<b>53</b>
<b>SEZNAM POUŽITÝCH VELIČIN .....</b>	<b>54</b>
<b>SEZNAM PŘÍLOH .....</b>	<b>55</b>
<b>PŘÍLOHA 1 – ZDROJOVÝ TEXT FUNKCÍ KNIHOVNY .....</b>	<b>56</b>

## ÚVOD

Cílem bakalářské práce je vytvoření knihovny funkcí základních termodynamických výpočtů s ideálními plyny a se směsmi ideálních plynů v programu MATLAB. Základními výpočty rozumíme výpočty od převodů teploty mezi dvěma v praxi nejčtenějšími stupnicemi, přes určení vlastností pracovní látky, výpočet stavových veličin v krajních stavech termodynamického děje, až například po určení práce vykonané (příp. spotřebované) během děje nebo cyklu. Jednoduše lze říci, že jde o výpočty s ideálními plyny prováděné v rámci předmětu termomechanika.

S běžnými termodynamickými výpočty je často svázáno grafické znázornění termodynamické změny v  $p$ - $V$  či  $T$ - $S$  diagramu. Proto by v této práci měli být vytvořeny i funkce pro vykreslení těchto diagramů.

Hlavním účelem vytvoření knihovny funkcí je usnadnění a urychlení práce při výpočtech termodynamických dějů a oběhů s ideálními plyny.

Součástí práce je také popis funkcí a výpočtů. Vzhledem k počtu funkcí a k celkovému rozsahu jejich zdrojového textu by detailní popis těchto funkcí byl zdlouhavý a pravděpodobně i nemožný. V práci jsou proto popsány pouze základní pojmy termodynamiky, ze kterých výpočty ve funkcích knihovny vycházejí, spolu se základními principy těchto výpočtů a se základními funkcemi MATLABu. Důležitější než detailní popis je vytvoření knihovny samotné tak, aby byla pro uživatele jednoduchá a zároveň účelná a užitečná.

# 1 MATLAB – ZÁKLADNÍ POJMY

## 1.1 Úvod k programu MATLAB

Program MATLAB® představila v roce 1984 americká společnost The MathWorks, Inc. Název MATLAB je zkratkou z anglického matrix laboratory (maticová laboratoř). Jde o počítačový program určený speciálně pro inženýrské a vědecké výpočty. Původně byl navržen pro vykonávání matematických výpočtů pomocí matic, o čemž vypovídá i samotný název programu. V průběhu let se ale vyvinul ve flexibilní výpočtový systém schopný řešit v podstatě jakýkoliv technický problém.

Jde o interaktivní prostředí pro vědeckotechnické výpočty, modelování a simulace, vizualizaci dat, vývoj algoritmů a dále pro datové rozborů a numerické výpočty. Prostřednictvím MATLABu lze řešit technické výpočetní problémy rychleji než prostřednictvím tradičních programovacích jazyků, jako jsou například C, C++, Pascal apod. MATLAB je rozsáhlý program nabízející neuvěřitelně širokou škálu funkcí, o čemž svědčí i fakt, že již základní verze programu nabízí velmi rozsáhlou knihovnu předdefinovaných funkcí (více než 1000 funkcí) k usnadnění a zefektivnění technických programovacích úkolů a umožňuje propojení s jinými aplikacemi a programy. Tím jsou výrazně rozšířeny možnosti využití aplikací a algoritmů vytvořených v tomto programu. Oproti jiným programovacím jazykům tedy nabízí mnohem rozšířenější škálu použití. Navíc i přes tento fakt je jeho používání poměrně jednoduché. MATLAB lze použít v širokém spektru aplikací v nejrůznějších oborech, včetně zpracování signálu, obrazu, komunikaci, návrhu řízení, zkoušení a měření, v neposlední řadě také finančního modelování, finanční analýze a také výpočetní biologii. [1] [2]

Přestože je program MATLAB používán více než milionem inženýrů, vědců a techniků po celém světě v širokém spektru oborů a v posledních letech vychází každoročně nová aktualizace programu rozšířená o další vestavěné funkce a o nové možnosti propojení s jinými programy, nebyla doposud vytvořena část knihovny funkcí tohoto programu, která by byla určena pro výpočty v tak běžné části fyziky, jakou je termodynamika.

## 1.2 Základní pojmy programovacího prostředí MATLAB

### **Skript:**

Skript je typ tzv. m-souboru (soubor vytvořený v programu MATLAB s příponou \*.m). Tento soubor obsahuje v logickém sledu příkazy MATLABu a různé povely, které po zavolání (spuštění) provede. [3]

### **Funkce:**

Funkce je stejně jako skript druh m-souboru. Rovněž obsahuje zdrojový text, ve kterém jsou popsány operace, jež budou provedeny po jejím zavolání. Ve zdrojovém textu je funkce uvozena slovem *function*. Na rozdíl od skriptu může mít funkce jeden či více vstupních a výstupních parametrů, jejichž počet může být buď stálý, nebo proměnný. Od charakteru vstupních a výstupních parametrů se odvíjí způsob volání funkcí. Funkce může být volána i v rámci jiné funkce. Blíže jsou způsoby tvorby zdrojového textu a způsoby volání různých druhů funkcí popsány v [3].

## Knihovna funkcí:

Pod pojmem knihovna funkcí rozumíme skupinu funkcí, které jsou spolu nějakým způsobem svázány, např. účelem, ke kterému slouží. Knihovny funkcí slouží uživateli většinou k usnadnění programování, v případě naší knihovny k usnadnění programování termodynamických výpočtů.

### 1.3 Funkce MATLABu použité v bakalářské práci

V této podkapitole je stručně vysvětlena funkce povelů, zabudovaných příkazů a operátorů programu MATLAB, které jsou použity při tvorbě knihovny funkcí základních termodynamických výpočtů.

#### Základní operátory:

Tab. 1.1 Výpis základních operátorů [4]

Příkaz	Příklad použití příkazu	Význam příkazu
+	1+2	součet
-	25-16	rozdíl
*	3*4	součin
/	36/3	podíl
^	2^3	umocnění (dvě na třetí)

#### Relační operátory:

Tab. 1.2 Výpis relačních operátorů [4]

Příkaz	Význam příkazu
==	rovno
>	větší
<	menší
>=	větší nebo rovno
<=	menší nebo rovno
~=	nerovná se

Výstupem použití relačních operátorů je 1 v případě, že relace platí, nebo 0 v případě, že relace neplatí. Relační operátory lze použít jak mezi proměnnými obsahujícími číselné hodnoty, tak mezi proměnnými obsahujícími textové řetězce, případně mezi hodnotami nebo řetězci samotnými.

## Logické operátory:

Tab. 1.3 Výpis logických operátorů [4]

Příkaz	Význam příkazu
&	logická spojka A
	logická spojka NEBO
not	negace

Výsledkem logické operace je 1 nebo 0.

## Práce se znakovými řetězci:

Tab. 1.4 Výpis příkazů pro práci se znakovými řetězci [4]

Příkaz	Příklad použití příkazu	Význam příkazu
zápis znakové proměnné	a='vkladany text'	do proměnné a se vloží text uvedený mezi apostrofy
disp	disp('Ahoj')	zobrazí na displej text uvedený mezi apostrofy
ischar	ischar(promenna)	zjištění, zda je proměnná typu text, v tom případě je výsledek 1, jinak je výsledek 0
strcmp	strcmp(text1,text2)	srovnání dvou proměnných obsahujících textové řetězce, výsledek je 0, pokud řetězce nejsou zcela shodné nebo 1 v případě, že jsou shodné
strncmp	strncmp(text1,text2,n)	totéž co strcmp, ale srovnáváme pouze prvních n znaků řetězců

## 2D grafy:

Tab. 1.5 Výpis příkazů pro práci s 2D grafy [4]

Příkaz	Příklad použití příkazu	Význam příkazu
plot	x=0:pi/50:5*pi	definice vektoru osy x
	plot(x,sin(x))	vykreslení spojitého 2D grafu
hold	hold on	další graf (příp. popisek) je přidán do aktuálního grafického okna
	hold off	další graf (příp. popisek) přepíše aktuální graf
axis	axis([xmin,xmax,ymin,ymax])	nastaví osy v zadaných mezích
text	text(x,y,'retezec')	vepíše do grafu text "retezec" na pozici x, y
title	title('retezec')	vepíše do grafu název (doprostřed nahoru)
xlabel	xlabel('retezec')	popíše osu x
ylabel	ylabel('retezec')	popíše osu y

**Ostatní příkazy:**

Tab. 1.6 Výpis ostatních příkazů [3], [4]

Příkaz	Příklad použití příkazu	Význam příkazu
input	a=input('Zadejte hodnotu a: ')	po zobrazení výzvy vepsané mezi aspostrify na displej čeká na vložení hodnoty (text, číslo) do proměnné, zadání se potvrzuje klávesou ENTER
linspace	vektor=linspace(od,do,počet)	naplní vektor s krajními hodnotami <i>od</i> a <i>do</i> tak, že prvky tvoří aritmetickou řadu a celkový počet prvků je <i>počet</i>
nargin	pvp=nargin	do proměnné <i>pvp</i> vloží počet vstupních parametrů, použití u funkce s proměnným počtem vstupních parametrů
varargin	vstup=varargin	uloží do pole buněk s názvem <i>vstup</i> všechny vstupní parametry, použití u funkce s proměnným počtem vstupních parametrů

Ve funkcích knihovny se dále vyskytují programátorské příkazy *if*, *switch*, *case*, *for* a *break*. Tyto příkazy jsou však běžné a obecně známé, proto se jimi v této části zabývat nebudeme. Příklady použití těchto příkazů lze nalézt např. v [3].

## 2 ZÁKLADNÍ FUNCE

Při tvorbě knihovny funkcí pro základní termodynamické výpočty vycházíme z předpokladu, že na počátku svého výpočtu zná uživatel druh pracovní látky, tj. v našem případě druh ideálního plynu nebo směsi, se kterou dále pracuje. Od druhu pracovní látky se pak odvíjí základní termodynamické vlastnosti a veličiny potřebné pro další výpočty. V této kapitole je vedle funkce volby a funkce vlastností pracovní látky také popsána funkce stavové rovnice spolu s funkcemi pro převod tlaků a teplot. Zdrojový text všech funkcí uvedených v této kapitole i ve všech následujících kapitolách je součástí přílohy.

### 2.1 Volba pracovní látky

#### Základní pojmy a veličiny:

Hovoříme-li o ideálním plynu, myslíme tím plyn, jehož molekuly jsou od sebe tak vzdáleny, že potenciální energie mezi molekulami je zanedbatelná. Co se týče působení konzervativních sil, jsou na sebe částice ideálního plynu vzájemně nezávislé, ovlivňují se pouze během dokonale pružných srážek, ke kterým dochází během jejich tepelného pohybu. Ideální plyn je dokonale stlačitelný, bez vnitřního tření a dále také uvažujeme, že fyzikální vlastnosti ideálního plynu zůstávají s měněním se tlakem a teplotou konstantní. Pro ideální plyn pak platí stavová rovnice ideálního plynu.

Molární hmotnost  $M$  je fyzikální veličina udávající hmotnost 1 molu dané látky, v našem případě zvoleného plynu. Molární hmotnost lze vypočítat ze vztahu:

$$M = \frac{m}{n} \quad [\text{kg} \cdot \text{kmol}^{-1}] \quad (2.1)$$

kde  $n$  je látkové množství plynu a  $m$  hmotnost plynu. Někdy je však jednodušší určit molární hmotnost pomocí relativní molekulové hmotnosti  $M_r$ , případně relativní atomové hmotnosti  $A_r$ , jejíž hodnotu obvykle získáme jednoduchým výpočtem, v případě relativní atomové hmotnosti pak přímo z periodické tabulky prvků. Relativní molekulová (atomová) hmotnost vyjadřuje, kolikrát je hmotnost molekuly větší než  $1/12$  klidové hmotnosti atomu uhlíku  $^{12}_6\text{C}$ . Z této definice pak vychází vztah pro výpočet relativní molekulové hmotnosti.

Pro  $M_r$  dále platí vztah:

$$M_r = \frac{m_0}{m_u} \quad [-] \quad (2.2)$$

kde  $m_0$  je hmotnost molekuly a  $m_u$  atomová hmotnostní konstanta ( $m_u = 1,66054 \cdot 10^{-27}$  kg). Obě tyto veličiny mají stejnou jednotku. Z toho plyne, že  $M_r$  je bezrozměrné číslo.

Molární hmotnost lze také určit ze vztahu (2.3):

$$M = m_0 \cdot N_A \quad [\text{kg} \cdot \text{kmol}^{-1}] \quad (2.3)$$

kde  $m_0$  je hmotnost jedné částice (molekuly) a  $N_A$  je Avogadrova konstanta ( $N_A = 6,02214 \cdot 10^{23} \text{ mol}^{-1}$ ). Ze vztahů (2.2) a (2.3) můžeme určit vztah mezi molární hmotností a relativní molekulovou hmotností. Dosazením  $m_0$  z jedné rovnice do druhé a úpravou získáme vztah pro  $M$ :

$$M = M_r \cdot m_u \cdot N_A \quad [\text{kg} \cdot \text{kmol}^{-1}] \quad (2.4)$$

Po dosazení číselných hodnot konstant  $m_u$  a  $N_A$  získáme konečnou podobu vztahu mezi  $M$  a  $M_r$ :

$$M = M_r \cdot 1,66054 \cdot 10^{-27} \cdot 6,02214 \cdot 10^{23} \\ M \doteq M_r \cdot 10^{-3} \quad [\text{kg} \cdot \text{mol}^{-1}] \quad (2.5)$$

Pokud nyní vezmeme v úvahu převod jednotek z molů na kilomoly, pak platí, že:

$$M \doteq M_r \quad [\text{kg} \cdot \text{kmol}^{-1}] \quad (2.6)$$

Poissonova konstanta  $\kappa$  je poměr mezi měrnou tepelnou kapacitou za konstantního tlaku  $c_p$  a měrnou tepelnou kapacitou za konstantního objemu  $c_v$ :

$$\kappa = \frac{c_p}{c_v} \quad [-] \quad (2.7)$$

Veličiny  $c_p$  a  $c_v$  mají stejnou jednotku, proto je Poissonova konstanta bezrozměrné číslo. Přibližnou hodnotu Poissonovy konstanty lze určit podle počtu atomů v molekule plynu. Pro jednoatomové plyny je přibližná hodnota  $\kappa = 1,67$ , pro plyny tvořící dvouatomové molekuly je pak  $\kappa = 1,41$  a v případě tříatomových molekul plynů je  $\kappa = 1,30$ . Chceme-li však zpřesnit naše výpočty s touto konstantou, je třeba použít její tabulkové hodnoty pro námi zvolený plyn.

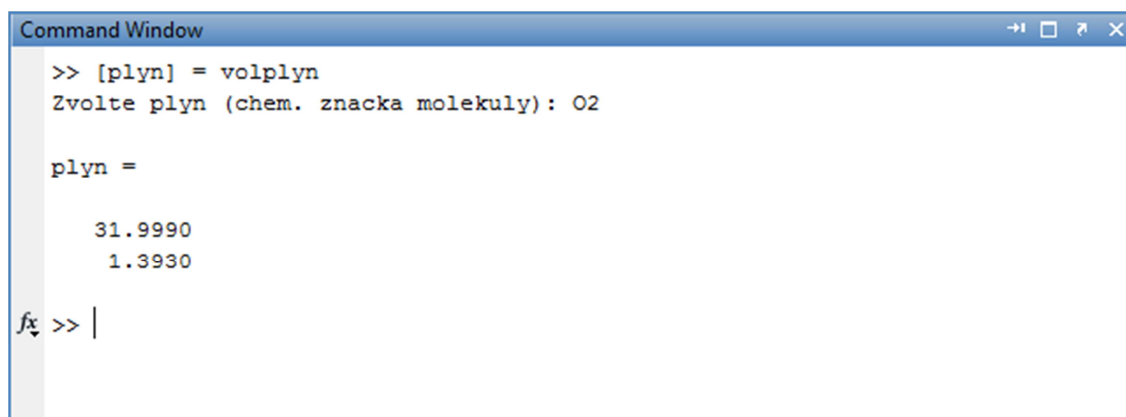
### Funkce volby pracovní látky:

Funkce sloužící uživateli ke zvolení druhu pracovní látky, jíž je ideální plyn, popřípadě směs ideálních plynů, nese název *volplyn*. Uživatel má dispozici několik základních ideálních plynů a směsí ideálních plynů nejčastěji užívaných v termodynamických výpočtech. Volbou druhu plynu uživatel získá dvě hodnoty, a sice tabulkovou hodnotu molární hmotnosti  $M$  a Poissonovy konstanty  $\kappa$  molekuly zvoleného plynu při teplotě 25 °C a tlaku 100 kPa [5]. Tyto hodnoty pak spolu ve sloupcovém vektoru, kde na prvním řádku je hodnota  $M$  a na druhém řádku hodnota  $\kappa$ , tvoří výstupní parametr funkce *volplyn*. Vstupní parametry tato funkce nemá.

## Volání funkce:

Funkce má pouze výstupní parametr, který je zpravidla potřebný pro další výpočty. Je proto výhodné funkci volat „do proměnné“, odkud pak můžeme hodnoty použít. Samotná volba plynu spočívá v zapsání chemického vzorce molekuly plynu, v případě zvolení vzduchu pak zapsáním slova „vzduch“ poté, co je k tomu uživatel vyzván pokynem zobrazeným na displeji. Vložení chemického vzorce molekuly se potvrzuje tlačítkem enter.

Na obr. 2.1 můžeme vidět příklad použití funkce *volplyn* při volbě kyslíku, jakožto pracovní látky.



```

Command Window
>> [plyn] = volplyn
Zvolte plyn (chem. znacka molekuly): O2

plyn =

    31.9990
     1.3930

fx >> |
  
```

Obr. 2.1 Příklad použití funkce *volplyn*

## 2.2 Vlastnosti plynu

### Základní pojmy a veličiny:

Měrná plynová konstanta  $r$  je konstanta užívaná zejména při výpočtech ze stavové rovnice ideálního plynu. Určíme ji pomocí vztahu:

$$r = \frac{R_m}{M} \quad [\text{J} \cdot \text{kg}^{-1} \cdot \text{K}^{-1}] \quad (2.8)$$

kde  $R_m$  je univerzální plynová konstanta, a  $M$  molární hmotnost molekuly ideálního plynu. Hodnota univerzální plynové konstanty je  $R_m = 8314,472 \text{ J} \cdot \text{kmol}^{-1} \cdot \text{K}^{-1}$ . Pokud bychom pro výpočet požadovali přesnější hodnotu, vyjdeme z toho, že univerzální plynová konstanta je dána součinem Avogadrovy a Boltzmannovy konstanty. Platí tedy, že:

$$R_m = N_A \cdot k \quad [\text{J} \cdot \text{kmol}^{-1} \cdot \text{K}^{-1}] \quad (2.9)$$

Po dosazení tabulkových hodnot obou zmíněných konstant bychom získali přesnou hodnotu univerzální plynové konstanty. Pro běžné termodynamické výpočty je však hodnota  $R_m$  uvedená výše dostatečně přesná. Hodnotu molární hmotnosti  $M$  získáme jako část vstupního parametru.

Měrná tepelná kapacita je teplo potřebné pro ohřev 1 kg látky o 1 K (1 °C). U plynů rozlišujeme měrnou tepelnou kapacitu za konstantního tlaku  $c_p$  a měrnou tepelnou kapacitu za konstantního objemu  $c_v$ . Pro tyto veličiny platí vztah (2.7).

Mezi veličinami  $r$ ,  $c_p$  a  $c_v$  platí tzv. Mayerův vztah – (2.10). Tento vztah je odvozený z první formy prvního zákona termodynamiky a ze stavové rovnice v diferenciálním tvaru. Odvození zde provádět nebudeme, uvedeme pouze finální podobu tohoto vztahu:

$$c_p = r + c_v \quad [\text{J} \cdot \text{kg}^{-1} \cdot \text{K}^{-1}] \quad (2.10)$$

### Funkce k určení vlastností plynu:

K určení základních termodynamických vlastností zvolené pracovní látky, konkrétně k určení měrné plynové konstanty  $r$ , měrné tepelné kapacity za konstantního tlaku  $c_p$  a měrné tepelné kapacity za konstantního objemu  $c_v$ , slouží uživateli v knihovně funkcí termodynamických výpočtů funkce s názvem *vlastn*.

### Princip výpočtů ve funkci:

Výpočty ve funkci vycházejí ze znalosti vztahů (2.7), (2.8) a (2.10). Z těchto vztahů je zřejmé, že určované vlastnosti plynů jsou závislé na hodnotě  $M$  a  $\kappa$ , tedy na vlastnostech, které přímo souvisí s druhem plynu. Tyto hodnoty tvoří vstupní parametr funkce.

Pokud se opět podíváme na vztahy uvedené výše, můžeme prohlásit, že ze vztahu (2.8) lze přímo určit hodnotu  $r$ , neboť obě hodnoty potřebné k jejímu výpočtu známe.

Po určení velikosti  $r$  tvoří vztahy (2.8) a (2.10) soustavu dvou rovnic o dvou neznámých. Tuto soustavu vyřešíme např. dosazovací metodou následovně:

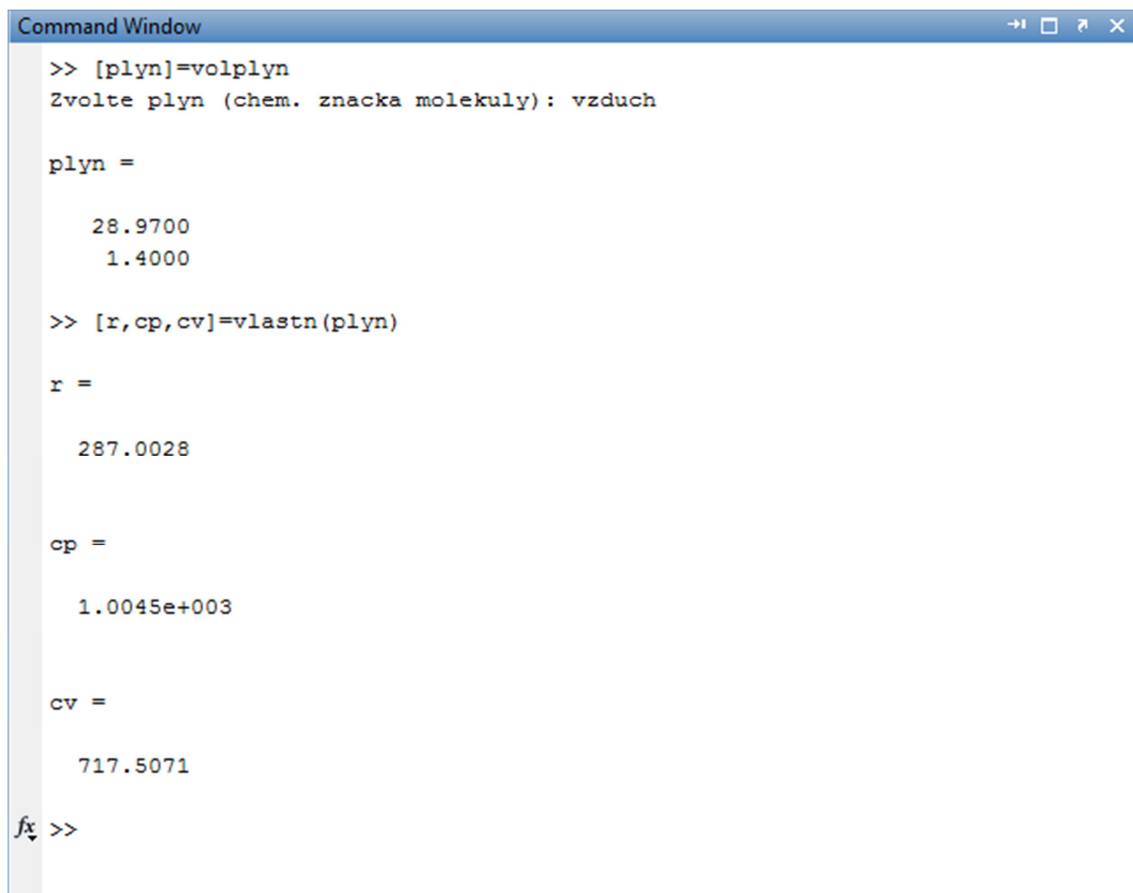
$$\begin{array}{l} \kappa = \frac{c_p}{c_v} \Rightarrow c_v = \frac{c_p}{\kappa} \\ \hline c_p = r + c_v \\ \hline c_p = r + \frac{c_p}{\kappa} \Rightarrow c_p = \frac{r}{1 - \frac{1}{\kappa}} = \frac{\kappa \cdot r}{\kappa - 1} \end{array}$$

Hodnotu Poissonovy konstanty známe (je součástí vstupního parametru funkce), rovněž v tomto okamžiku známe hodnotu měrné plynové konstanty. Řešením soustavy rovnic jsou vyjádřené vztahy pro  $c_p$  a  $c_v$ . Nejprve číselně určíme velikost  $c_p$ , tu dosadíme do vztahu pro  $c_v$  a konečně i tuto hodnotu číselně vyjádříme. Nyní jsme získali všechny hledané hodnoty.

## Volání funkce:

Funkce má jeden vstupní a tři výstupní parametry. Vstupním parametrem je sloupcový vektor, kde se na prvním řádku nachází hodnota molární hmotnosti  $M$  [ $\text{kg}\cdot\text{kmol}^{-1}$ ] a na druhém řádku hodnota Poissonovy konstanty  $\kappa$  [-] molekuly plynu, se kterým pracujeme. Tyto hodnoty jsou tedy uspořádány tak, jako je tomu u výstupního parametru funkce volby pracovní látky (funkce *volplyn*), čímž je zajištěna návaznost těchto dvou funkcí. Výstupem funkce jsou vypočítané hodnoty vlastností plynu  $r$ ,  $c_p$  a  $c_v$ . Veličiny mají jednotku  $\text{J}\cdot\text{kg}^{-1}\cdot\text{K}^{-1}$ .

Příklad použití funkce *vlastn* spolu s funkcí *volplyn*, ve kterém jsou určovány vlastnosti vzduchu, můžeme vidět na následujícím obrázku (obr. 2.2).



```
Command Window
>> [plyn]=volplyn
Zvolte plyn (chem. znacka molekuly): vzduch

plyn =

    28.9700
    1.4000

>> [r,cp,cv]=vlastn(plyn)

r =

    287.0028

cp =

    1.0045e+003

cv =

    717.5071

fx >>
```

Obr. 2.2 Určení základních vlastností vzduchu

## 2.3 Stavová rovnice

### Charakteristika pojmů a veličin:

Stavová rovnice ideálního plynu je rovnice vyjadřující vztah mezi stavovými veličinami pracovní látky (ideálního plynu) v jednom konkrétním stavu a její základní tvar je následující:

$$p \cdot V = m \cdot r \cdot T \quad (2.11)$$

V případě měrných veličin lze použít také tvar:

$$p \cdot v = r \cdot T \quad (2.12)$$

kde  $p$  je tlak plynu,  $V$  objem plynu,  $v$  měrný objem plynu,  $m$  hmotnost plynu,  $r$  měrná plynová konstanta a  $T$  termodynamická teplota plynu.

### Funkce stavové rovnice:

Funkce stavové rovnice má v knihovně funkcí název *str*. Slouží k určení jedné neznámé stavové veličiny, případně hmotnosti plynu, ze stavové rovnice.

### Princip výpočtů ve funkci:

Vycházíme z řešitelnosti stavové rovnice. Řešitelnost je zajištěna dodržením maximálního počtu neznámých. Nesmí zde existovat více než jedna neznámá. Před samotným výpočtem ve funkci nejprve zjistíme, která veličina je touto neznámou. Poté vyjádříme vztah pro její výpočet a určíme její hodnotu.

### Volání funkce:

Funkce *str* je funkce s proměnným počtem vstupních parametrů. Vstupní parametry mohou být 4 nebo jich může být 5, jiný počet přípustný není. V případě zadání 4 vstupních parametrů se automaticky pro další výpočty i pro výstup uvažuje hmotnost  $m = 1$  kg.

Výstupní parametry této funkce jsou číselné hodnoty stavových veličin  $p$ ,  $V$ ,  $T$  a hodnota hmotnosti pracovní látky  $m$ , přičemž platí, že hodnota, která byla na počátku neznámou, je na výstupu již známa.

Při volání funkce nejprve do hranatých závorek zapíšeme názvy proměnných, do kterých budou po provedení výpočtů funkce uloženy hodnoty výstupních parametrů v pořadí  $p$ ,  $V$ ,  $T$  a  $m$ . Dále znaménko rovná se, název funkce (*str*) a do kulatých závorek vstupní parametry. Při zadávání hodnot vstupních parametrů je důležité dodržet jejich pořadí, rozměr (hodnota, vektor) a také jednotky dle tab. 2.1.

Tab. 2.1 Vstupní parametry funkce *str*

Pořadí	Veličina (proměnná)	Rozměr (hodnota, vektor)	Jednotka
1.	$p$	hodnota	[Pa]
2.	$V$	hodnota	[m <sup>3</sup> ]
nebo	$v$	hodnota	[m <sup>3</sup> /kg]
3.	$T$	hodnota	[K]
4.	$\text{plyn}=[M;\kappa]$	sloupcový vektor	$M$ [kg/kmol] $\kappa$ [-]
(5.)	$m$	hodnota	[kg]

Důležité je upozornění, že na rozdíl od všech následujících funkcí (vyjma funkce *apstav*) je hodnota tlaku zadávána v Pa. Funkce *str* je nejčastěji volána v jiných funkcích v průběhu výpočtů, ve kterých se tlak vyskytuje v Pa, proto je tento způsob zadávání tlaku vhodnější.

Pokud uživatel nezadá hodnotu hmotnosti nebo požadavek na její výpočet, musí ve svých výpočtech uvažovat to, že nepočítá s objemem  $V$ , nýbrž s měrným objemem  $v$  a zároveň se automaticky  $m = 1$  kg. Vektor *plyn* musí být zadán vždy číselně.

Zná-li uživatel při volání funkce číselnou hodnotu vstupního parametru, pak na její místo (dle tab. 2.1) zadá tuto číselnou hodnotu v požadované formě a jednotkách, případně název proměnné, která tuto hodnotu obsahuje. Jestliže je některý ze vstupních parametrů pro uživatele neznámou, pak při volání funkce uživatel na její místo zadá libovolný textový řetězec. Textový řetězec je text uzavřený mezi apostrofy, např.: 'textový řetězec'. Textový řetězec však v případě této funkce smí být zadán pouze na místě jednoho vstupního parametru (vyplývá z řešitelnosti funkce).

Ke zjištění zda proměnná obsahuje textový řetězec a je tedy neznámou, v této i ve všech následujících funkcích příkazu *ischar*, jehož funkce je popsána v tab. 1.4.

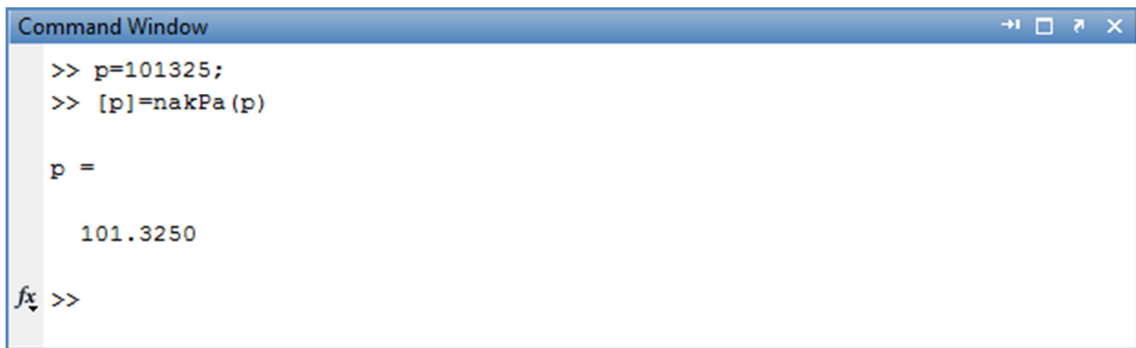
## 2.4 Převody veličin

### 2.4.1 Převody tlaku z Pa na kPa a naopak

Pro měření tlaku je obvyklejší jednotkou kPa, neboť 1 Pa je jednotkou poměrně malou. Veškeré výpočty v termodynamice se však provádějí v základních jednotkách, tedy v Pa. Tlak se vyskytuje téměř ve většině funkcí této knihovny, proto je pro převody tlaku výhodné zavést zvláštní funkce.

Pro převod tlaku z kPa na Pa slouží funkce *naPa*, pro převod tlaku z Pa na kPa pak funkce *nakPa*. Obě funkce mají jeden vstupní a jeden výstupní parametr. Vstupním parametrem může být číselná hodnota tlaku nebo textový řetězec. Textový řetězec je ve funkcích termodynamických dějů používán při volání funkce k označení neznámé proměnné.

Je-li vstupním parametrem číselná hodnota tlaku, pak je výstupním parametrem rovněž hodnota tlaku, avšak přepočítaná na jednotky odpovídající názvu funkce. Je-li vstupním parametrem textový řetězec, pak je výstupním parametrem opět tentýž textový řetězec. Převod se provede pouze pro číselnou hodnotu vstupního parametru.



```
Command Window
>> p=101325;
>> [p]=nakPa(p)

p =

    101.3250

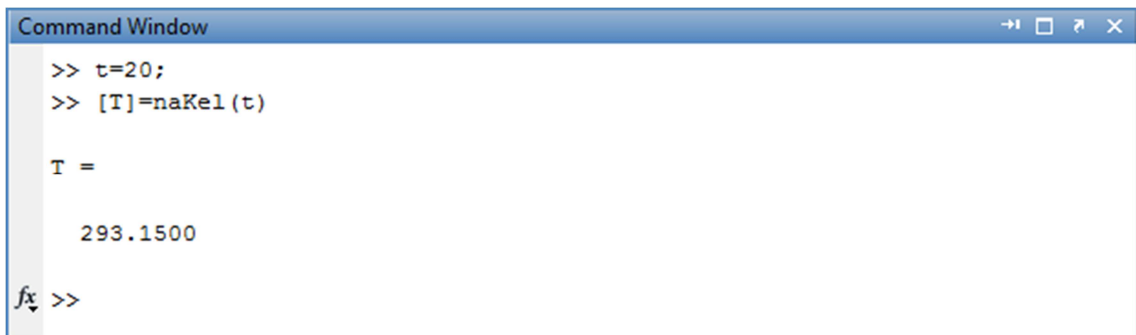
fx >>
```

Obr. 2.3 Příklad použití funkce *nakPa*

## 2.4.2 Převody teploty ze °C na K a naopak

V případě teploty je tomu podobně jako u tlaku. Ve většině výpočtů v termodynamice se počítá s hodnotami teplot v kelvinech, avšak pro měření teploty jsou obvyklejší jednotkou stupně Celsia. Z toho důvodu je výhodné mít k dispozici funkce pro převod těchto jednotek.

V knihovně funkcí k převodu teploty ze stupňů Celsia na kelviny slouží funkce *naKel*, k převodu z kelvinů na stupně Celsia pak funkce *naCel*. Pro vstupní a výstupní parametry těchto funkcí platí stejné zákonitosti jako v případě převodů tlaku (podkapitola 2.4.1). Na obr. 2.4 potom můžeme vidět příklad použití funkce *naKel* pro převod teploty 20 °C na kelviny.



```
Command Window
>> t=20;
>> [T]=naKel(t)

T =

    293.1500

fx >>
```

Obr. 2.4 Příklad použití funkce *naKel*

## 3 TERMODYNAMICKÉ DĚJE

### 3.1 Základní teorie termodynamických dějů

Termodynamický děj je posloupnost stavů, kterými soustava prochází během termodynamického procesu. Tento děj probíhá mezi dvěma krajními stavy, které jsou obvykle očíslovány vzestupně ve směru probíhajícího děje. Pokud je pracovní látkou ideální plyn, je termodynamický děj vždy vratný. Termodynamický děj představuje v  $p$ - $V$  nebo  $T$ - $S$  diagramu křivku (posloupnost stavů) mezi dvěma body (krajními stavy). Mezi dvěma stejnými stavy je křivka pro různé děje odlišná (např. přímka, hyperbola, parabola).

Pro každý termodynamický děj je v knihovně funkcí základních termodynamických výpočtů zavedena zvláštní funkce. Každý děj je reprezentován prostřednictvím čtyř vektorů. Jde o vektory tlaku  $p$ , objemu  $V$  (popř. měrného objemu  $v$ ), teploty  $T$  a entropie  $S$  (popř. měrné entropie  $s$ ). Pro hodnoty na odpovídajících si pozicích v těchto vektorech platí stavová rovnice ideálního plynu a rovnice konkrétního termodynamického děje. Tyto vektory jsou pak výhodně používány v dalších výpočtech a funkcích, ať už pro určení energetických veličin, či vykreslení  $p$ - $V$  a  $T$ - $S$  diagramů apod.

### 3.2 Názvy funkcí

Názvy funkcí termodynamických dějů vycházejí ze zkratk vytvořených z názvů samotných termodynamických dějů. Názvy funkcí přiřazené konkrétním dějům jsou uvedeny v následující tabulce:

Tab. 3.1 Názvy funkcí termodynamických dějů

Termodynamický děj	Název funkce
Izobarický děj	ib
Izochorický děj	ich
Izotermický děj	it
Adiabatický děj	ad
Polytropický děj	pol

### 3.3 Princip výpočtů ve funkcích

#### 3.3.1 Izobarický, izochorický a izotermický děj

Jak je psáno podkapitole 3.1, termodynamické děje probíhají mezi dvěma krajními stavy. Pro každý z těchto stavů lze psát stavovou rovnici. Tímto získáme dvě pro každý děj jiné rovnice, které tvoří základní soustavu rovnic potřebnou pro vyřešení konkrétního termodynamického děje. Stavové rovnice krajních stavů konkrétních dějů jsou uvedeny v kapitole 4.

### Podmínky řešitelnosti:

Aby byla základní soustava rovnic řešitelná, nesmí být počet neznámých větší než 2. Hodnotu měrné plynové konstanty  $r$  známe u všech funkcí termodynamických dějů (dopočítává se v každé funkci ze vstupního parametru pomocí funkce *vlastn*). Dále nesmí nastat žádná ze tří pro každou funkci specifických kombinací neznámých. Tyto kombinace jsou pro každý děj rovněž uvedeny v kapitole 4. Je-li proměnná, která je u daného děje konstantní (izobarický děj – tlak  $p$ , izochorický děj – objem  $V$  (resp.  $v$ ), izotermický děj – teplota  $T$ ), nebo hodnota hmotnosti  $m$  neznámou veličinou, pak musí alespoň v jedné z rovnic soustavy být známy hodnoty všech ostatních proměnných.

Pokud není jedna z podmínek uvedených v tomto odstavci splněna, funkce další výpočty neprovede.

### Určení výstupních vektorů:

Po ověření platnosti podmínek řešitelnosti ve funkci nejprve zjistíme, které hodnoty jsou neznámé, a pomocí vhodně použité stavové rovnice (funkce *str*) postupně tyto neznámé určíme. Po výpočtu stavových veličin  $p$ ,  $V$  (příp.  $v$ ) a  $T$  v obou krajních stavech naplníme hodnotami výstupní vektory.

Pro určení hodnot výstupního vektoru stavové veličiny, jenž je v případě konkrétního děje konstantní, lze použít funkci *linspace*. Vektor je naplněn 300 stejnými hodnotami. Funkci *linspace* lze použít i k určení hodnot jednoho z vektorů stavové veličiny, která u tohoto děje konstantní není. V tomto případě se vektor naplní opět 300 hodnotami, přičemž první z nich je hodnota dané stavové veličiny v jednom krajním stavu, poslední je hodnota dané veličiny v druhém krajním stavu a všechny hodnoty vektoru tvoří aritmetickou řadu. Třetí stavová veličina je u všech uvedených dějů závislá na předchozí stavové veličině (té, která není konstantní a její vektor je plněn pomocí funkce *linspace*). Krajní hodnoty vektoru této třetí stavové veličiny známe. K určení zbývajících hodnot vektoru je třeba použít cyklus založený na znalosti základní rovnice konkrétního termodynamického děje.

Nakonec stanovíme vektor entropie  $S$ , případně měrné entropie  $s$ . Entropie je stavová veličina sloužící k rozšířenějšímu popisu stavu látky, případně soustavy. Hlavně pak slouží k vykreslení  $T$ - $S$  diagramu. Tato veličina je vyjádřením druhého zákona termodynamiky a umožňuje nám posoudit směr vývoje termodynamických soustav a dějů. Entropii lze chápat jako míru neuspořádanosti systému, míru znehodnocení kvality nebo jako míru disipace látky či energie. [6]

Entropie  $S$  (příp. měrná entropie  $s$ ) je závislá na hodnotě teploty a tlaku. K určení jejího vektoru je proto třeba použít cyklus *for*, v tomto případě vycházející ze znalosti vztahu pro entropii v jednom daném stavu:

$$S_1 = m \cdot c_p \cdot \ln(T_1) - m \cdot r \cdot \ln(p_1) \quad [\text{J}] \quad (3.1)$$

V kapitole 4 je uvedeno, jakým způsobem je každý z výstupních vektorů stavových veličin u jednotlivých dějů určen.

### 3.3.2 Adiabatický a polytropický děj

I v případě těchto termodynamických dějů vyjdeme při řešení výpočtů ze znalosti stavových rovnic krajních stavů, mezi kterými děj probíhá (2 rovnice). Tyto rovnice mají pro oba děje stejný tvar.

$$p_1 \cdot V_1 = m \cdot r \cdot T_1 \quad (3.2)$$

$$p_2 \cdot V_2 = m \cdot r \cdot T_2 \quad (3.3)$$

Dále při řešení každého z dějů využijeme znalosti základní rovnice daného děje (1 rovnice). V případě adiabatického děje je to rovnice:

$$p_1 \cdot V_1^\kappa = p_2 \cdot V_2^\kappa \quad (3.4)$$

Pro případ polytropického děje pak rovnice:

$$p_1 \cdot V_1^n = p_2 \cdot V_2^n \quad (3.5)$$

kde  $n$  je polytropický exponent.

Stavové rovnice spolu s rovnicí děje tvoří základní soustavu rovnic potřebnou pro vyřešení konkrétního termodynamického děje.

#### Podmínky řešitelnosti:

Aby byla soustava řešitelná, nesmí být počet neznámých veličin větší než 3. Jak je již psáno v podkapitole 3.3.1, hodnotu  $r$  známe u všech funkcí termodynamických dějů. V rovnici adiabatického děje je hodnota Poissonovy konstanty  $\kappa$  vždy známá, je součástí povinného vstupního parametru (vektor plyn). V rovnici polytropického děje může být neznámou hodnota  $n$ . Je-li tomu tak, musí být zadány hodnoty dvou stavových veličin v obou krajních stavech, tedy hodnoty  $p_1, p_2, T_1$  a  $T_2$  nebo  $p_1, p_2, V_1$  a  $V_2$  nebo do třetice  $T_1, T_2, V_1$  a  $V_2$ . Hodnota polytropického exponentu je závislá na hodnotách stavových veličin, proto ji mezi neznámé veličiny nezahrnujeme, pouze ošetříme, aby byly známy hodnoty stavových veličin potřebných pro její případný výpočet. Pokud známe hmotnost  $m$  a počet neznámých je 3, je třeba, aby se tyto neznámé nenacházely všechny v jednom krajním stavu. Je-li hmotnost  $m$  neznámou, nesmí nastat žádná z kombinací neznámých  $p_1$  a  $p_2, V_1$  a  $V_2$  nebo  $T_1$  a  $T_2$ .

Jestliže není dodržena některá z uvedených podmínek řešitelnosti, funkce děje neprovede žádný.

#### Určení výstupních vektorů:

Pokud jsou splněny všechny podmínky řešitelnosti, provedeme zjištění, které proměnné jsou ve funkci neznámé. Pomocí vhodně použité funkce *str* a funkce *apstav* (princip její činnosti popsán v kapitole 4.6.1) pak tyto neznámé stanovíme. Po výpočtu všech stavových veličin v obou krajních stavech určíme výstupní vektory.

Všechny stavové veličiny jsou na sobě závislé. Výstupní vektor jedné ze stavových veličin určíme pomocí funkce *linspace*. Tento vektor bude obsahovat 300 hodnot, první hodnota je hodnota dané veličiny v jednom krajním stavu, poslední hodnota je hodnota dané veličiny v druhém krajním stavu a všechny hodnoty vektoru tvoří aritmetickou řadu. Pro stanovení ostatních výstupních vektorů stavových veličin je třeba použít cyklus *for*, rovnici konkrétního děje a případně stavovou rovnici, přičemž využijeme znalosti stavových veličin  $p$ ,  $V$  a  $T$  v obou krajních stavech a znalosti rovnice (3.1).

V případě funkcí adiabatického a polytropického děje určujeme výstupní vektor tlaku  $p$  pomocí funkce *linspace*, vektory objemu  $V$ , teploty  $T$  a entropie  $S$  s využitím cyklu *for*.

### 3.4 Volání funkcí

Jde o funkce s proměnným počtem vstupních parametrů. Uživatel do funkce termodynamického děje zadá požadované vstupní parametry určující krajní stavy konkrétního termodynamického děje a případně hmotnost pracovní látky  $m$ . Hodnota hmotnosti pracovní látky je nepovinná. Pokud není zadána, funkce automaticky uvažuje  $m = 1$  kg a musíme uvažovat výpočet ve formě měrných veličin.

Při zadávání vstupních parametrů je vždy důležité dodržet pořadí zadávaných veličin, rozměr (hodnota, vektor) a jednotku. Řídíme se přitom tabulkou vstupních parametrů funkce konkrétního děje, která je uvedena u funkce každého termodynamického děje v kapitole 4. V případě, že je veličina pro uživatele neznámou, zadá uživatel během volání funkce na její místo libovolný textový řetězec. Při zadávání neznámých parametrů je třeba zohlednit řešitelnost konkrétního termodynamického děje (podkapitoly 3.3.1 a 3.3.2). Dále musí být u všech funkcí zadána hodnota *plyn*, což je sloupcový vektor obsahující hodnotu molární hmotnosti zvoleného plynu a hodnotu Poissonovy konstanty (výstupní parametr funkce *volplyn*).

Na konci příkazu volání každé funkce termodynamického děje je vhodné psát středník, čímž se předejde zahlcení obrazovky. V opačném případě budou zobrazeny čtyři vektory, každý po třech stech hodnotách spolu s hodnotou hmotnosti.

Jako výstupní parametry uživatel získá čtyři řádkové vektory  $p$ ,  $V$ ,  $T$  a  $S$  (resp.  $p$ ,  $v$ ,  $T$ ,  $s$ ) představující konkrétní termodynamický děj a pokud je zadán požadavek na výpočet hmotnosti pracovní látky  $m$ , pak také tuto hmotnost.

### 3.5 Ošetření vstupů

Z podmínek řešitelnosti a ze základních matematických pravidel (např. dělení 0) vyplývají přípustné hodnoty vstupních proměnných pro výpočet. V každé funkci je třeba zajistit, aby ve vstupních parametrech byly pouze tyto přípustné hodnoty. Proto je nutné vstupy do funkcí takzvaně ošetřit. Ošetření provedeme tak, že prostřednictvím příkazů (nejčastěji *if*) zjistíme, zda je ve vstupním parametru, případně skupině parametrů, hodnota požadovaná pro výpočet, případně je-li vstupní parametr parametrem známým nebo neznámým. Pokud jsou dodrženy všechny požadované podmínky výpočtu, funkce po kontrole vstupních parametrů provede výpočet, v opačném případě výpočet funkce neprovede a do výstupních parametrů vloží textový řetězec 'Žádný výsledek'.

## 4 FUNKCE TERMODYNAMICKÝCH DĚJŮ

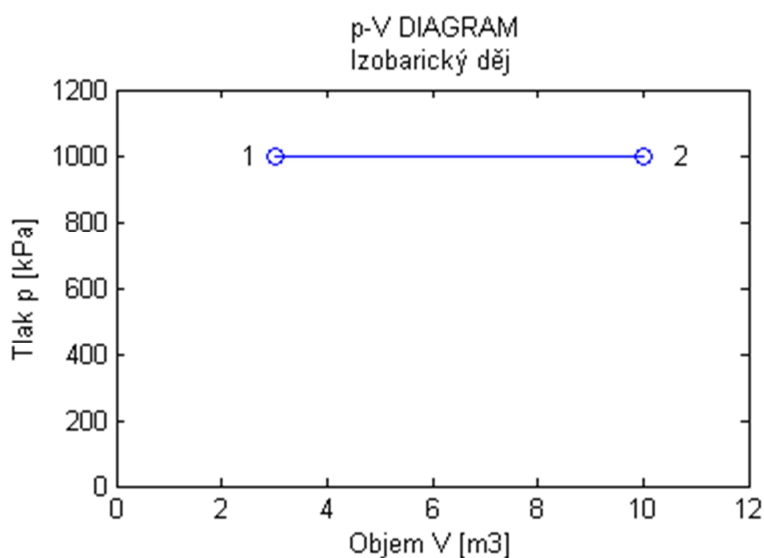
V této kapitole se budeme postupně zabývat všemi termodynamickými ději, funkcemi těchto dějů, způsobem určení výstupních vektorů u konkrétních funkcí a návody k jejich volání. Obecně pro všechny funkce termodynamických dějů byly tyto náležitosti popsány v podkapitolách 3.3 a 3.4, v této kapitole si uvedeme rozšíření těchto obecných pravidel u případů konkrétních termodynamických dějů.

### 4.1 Izobarický děj

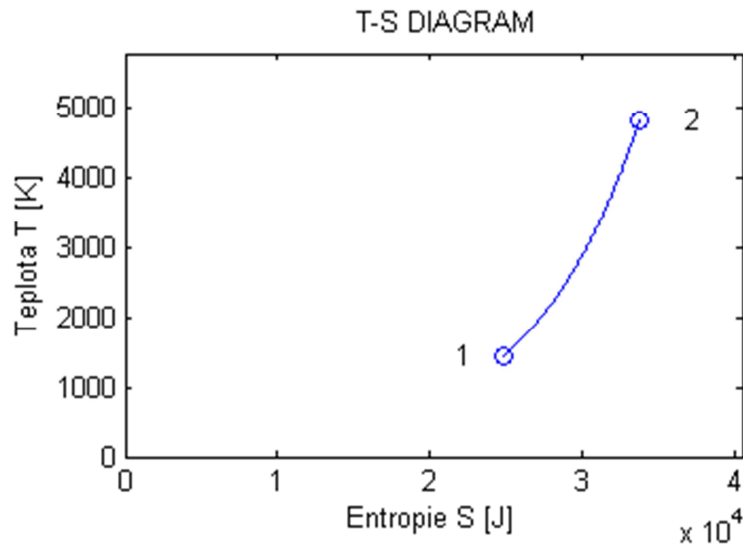
#### Základní pojmy a rovnice:

Izobarický děj je takový termodynamický děj, který probíhá za konstantního tlaku. V  $p$ - $V$  diagramu je znázorněn čarou rovnoběžnou s vodorovnou osou (osa  $V$ ) – viz obr. 4.1. Na obr. 4.2 pak můžeme vidět izobarický děj znázorněný v  $T$ - $S$  diagramu. Pro izobarický děj platí Gay-Lussacův zákon (4.1). [6]

$$\frac{V}{T} = konst. \quad (4.1)$$



Obr. 4.1  $p$ - $V$  diagram izobarického děje



Obr. 4.2 T-S diagram izobarického děje

Pro izobarický děj mezi stavy 1 a 2 podle Gay-Lussacova zákona platí rovnice ve tvaru:

$$\frac{V_1}{T_1} = \frac{V_2}{T_2} \quad (4.2)$$

Můžeme psát stavovou rovnici pro krajní stavy izobarického děje.

Pro stav 1:

$$\frac{V_1}{T_1} = \frac{m \cdot r}{p} \quad (4.3)$$

Pro stav 2:

$$\frac{V_2}{T_2} = \frac{m \cdot r}{p} \quad (4.4)$$

### Funkce izobarického děje *ib*:

Rovnice (4.3) a (4.4) tvoří základní soustavu rovnic pro izobarický děj. Aby byla zajištěna řešitelnost soustavy a tedy celé funkce, musí být dodrženy podmínky řešitelnosti (podkapitola 3.3.1). V případě izobarického děje nesmí nastat kombinace neznámých  $V_1$  a  $T_1$ ,  $V_2$  a  $T_2$  nebo  $p$  a  $m$ . Dále je-li neznámou  $p$  nebo  $m$ , pak alespoň v jedné z rovnic soustavy musí být známy hodnoty všech ostatních proměnných.

Tlak je v případě izobarického děje konstantní, výstupní vektor tlaku  $p$  se určí pomocí funkce *linspace* a bude obsahovat 300 stejných hodnot. Objem a teplota jsou na sobě závislé, jeden ze výstupních vektorů těchto stavových veličin stanovíme pomocí funkce *linspace* (v našem případě vektor objemu  $V$ ) a druhý pomocí cyklu *for* (vektor teploty  $T$ ).

Při volání funkce *ib* vycházíme z podkapitoly 3.4 a řídíme se níže uvedenou tabulkou:

Tab. 4.1 Vstupní parametry funkce *ib*

Pořadí	Veličina (proměnná)	Rozměr (hodnota, vektor)	Jednotka
1.	p	hodnota	[kPa]
2.	V1	hodnota	[m <sup>3</sup> ]
nebo	v1	hodnota	[m <sup>3</sup> /kg]
3.	T1	hodnota	[K]
4.	V2	hodnota	[m <sup>3</sup> ]
nebo	v2	hodnota	[m <sup>3</sup> /kg]
5.	T2	hodnota	[K]
6.	plyn=[M; κ]	sloupcový vektor	M [kg/kmol] κ [-]
(7.)	m	hodnota	[kg]

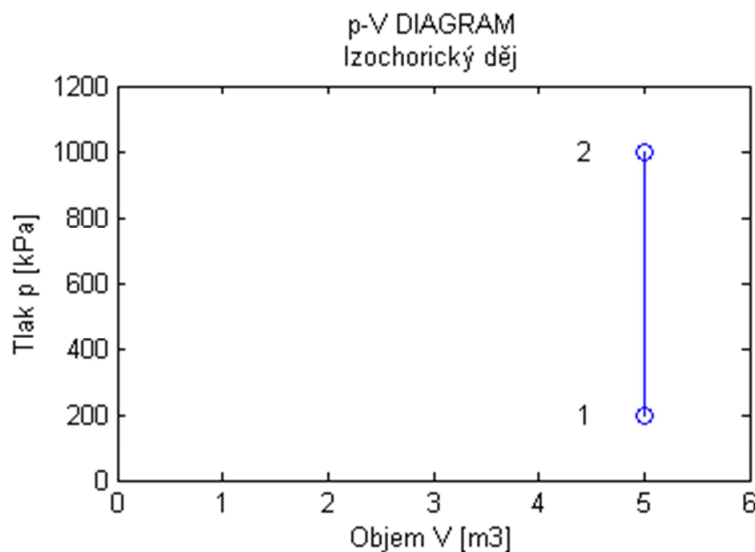
Zadání hmotnosti *m* je nepovinné. Nebude-li hmotnost zadána, funkce automaticky uvažuje *m* = 1 kg a počítá s měrnými veličinami. Ostatní vstupní parametry jsou povinné (musí být zadána číselná hodnota nebo textový řetězec).

## 4.2 Izochorický děj

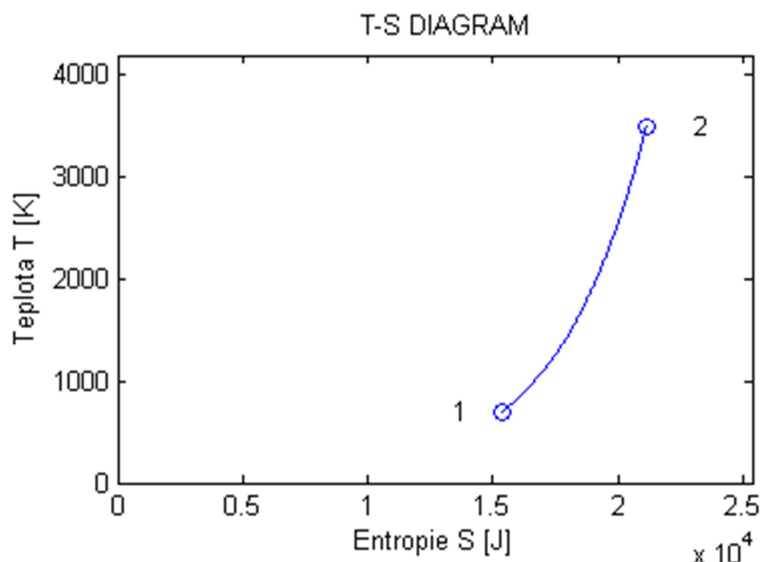
### Základní pojmy a rovnice:

Izochorický děj je termodynamický děj probíhající za konstantního objemu *V* (resp. měrného objemu *v*). V *p*-*V* diagramu je zastoupen čarou rovnoběžnou se svislou osou (osa *p*) – viz obr. 4.3. Na obr. 4.4 lze pak vidět izochorický děj zobrazený v *T*-*S* diagramu. Pro tento děj platí Charlesův zákon (4.5). [6]

$$\frac{p}{T} = konst. \quad (4.5)$$



Obr. 4.3 *p*-*V* diagram izochorického děje



Obr. 4.4 T-S diagram izochorického děje

Mezi stavy 1 a 2 platí pak při využití Charlesova zákona rovnice

$$\frac{p_1}{T_1} = \frac{p_2}{T_2} \quad (4.6)$$

Pro krajní stavy izochorického děje pak můžeme psát stavovou rovnici.

Pro stav 1:

$$\frac{p_1}{T_1} = \frac{m \cdot r}{V} \quad (4.7)$$

Pro stav 2:

$$\frac{p_2}{T_2} = \frac{m \cdot r}{V} \quad (4.8)$$

### Funkce izochorického děje *ich*:

Rovnice (4.7) a (4.8) tvoří základní soustavu rovnic pro vyřešení izochorického děje (podkapitola 3.3.1). Řešitelnost této soustavy a celé funkce je zajištěna dodržáním podmínek řešitelnosti stanovených v podkapitole 3.3.1. V případě funkce izochorického děje nesmí nastat kombinace neznámých  $p_1$  a  $T_1$ ,  $p_2$  a  $T_2$  nebo  $V$  a  $m$ . Pokud je neznámou  $V$  nebo  $m$ , musí být alespoň v jedné z rovnic soustavy známy hodnoty všech ostatních proměnných.

Izochorický děj je děj probíhající za konstantního objemu  $V$  (příp. měrného objemu  $v$ ), výstupní vektor objemu se tedy určí pomocí funkce *linspace* (300 stejných hodnot). Stavové veličiny tlak a teplota jsou na sobě závislé. Jeden z výstupních vektorů těchto veličin určíme pomocí funkce *linspace* (v případě funkce *ich* vektor tlaku  $p$ ) a druhý pomocí cyklu *for* (vektor teploty  $T$ ).

Při volání této funkce jako v předchozím případě vycházíme z podkapitoly 3.4 a řídíme se uvedenou tabulkou:

Tab. 4.2 Vstupní parametry funkce *ich*

Pořadí	Veličina (proměnná)	Rozměr (hodnota, vektor)	Jednotka
1.	V	hodnota	[m <sup>3</sup> ]
nebo	v	hodnota	[m <sup>3</sup> /kg]
2.	p1	hodnota	[kPa]
3.	T1	hodnota	[K]
4.	p2	hodnota	[kPa]
5.	T2	hodnota	[K]
6.	plyn=[M; κ]	sloupcový vektor	M [kg/kmol] κ[-]
(7.)	m	hodnota	[kg]

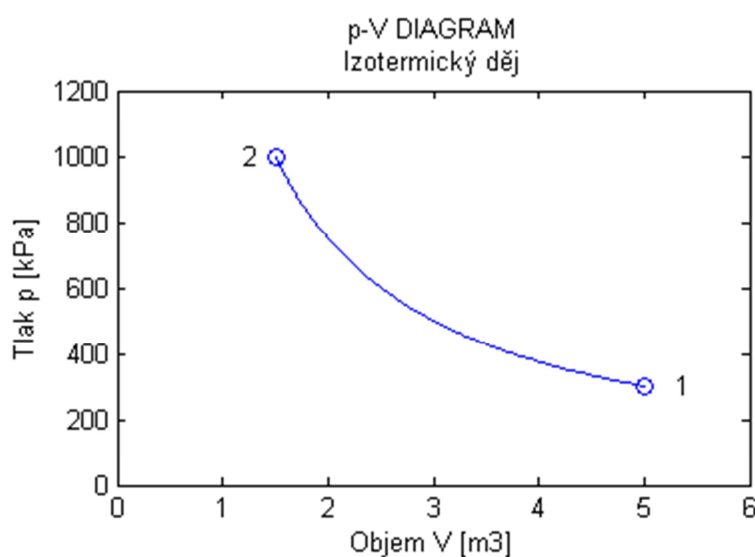
Zadání hmotnosti  $m$  je rovněž nepovinné. V případě, že nebude zadána, funkce automaticky uvažuje  $m = 1$  kg a počítá s měrnými veličinami. Zadání zbývajících parametrů je povinné (musí být zadána číselná hodnota nebo textový řetězec).

### 4.3 Izotermický děj

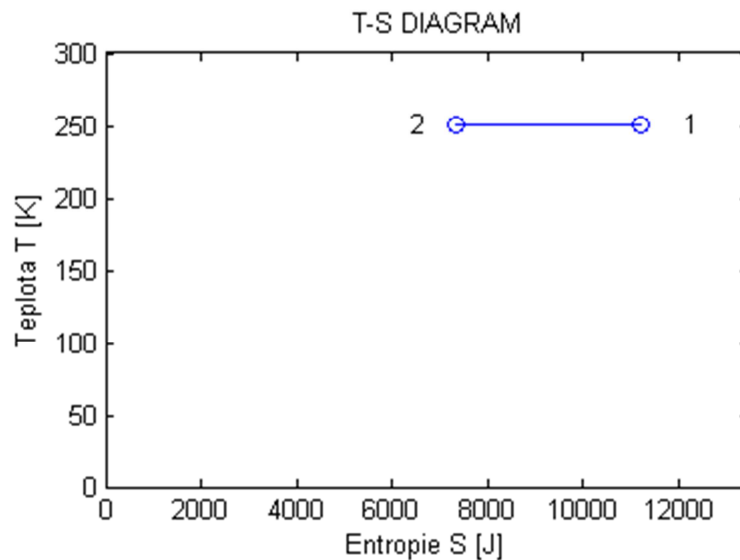
#### Základní pojmy a rovnice:

Termodynamický děj probíhající za konstantní teploty  $T$  nazýváme dějem izotermickým. Jak můžeme vidět na obr. 4.5 v  $p$ - $V$  diagramu je tento děj znázorněn rovnoosou hyperbolou.  $T$ - $S$  diagram tohoto děje můžeme vidět na obr. 4.6. Izotermický děj se řídí Boyle-Mariotteovým zákonem (4.9).

$$p \cdot V = konst. \quad (4.9)$$



Obr. 4.5  $p$ - $V$  diagram izotermického děje

Obr. 4.6  $T$ - $S$  diagram izotermického děje

Podle Boyle-Mariotteova zákona pro izotermický děj mezi stavy 1 a 2 platí rovnice ve tvaru:

$$p_1 \cdot V_1 = p_2 \cdot V_2 \quad (4.10)$$

Pro krajní stavy izotermického děje můžeme psát stavovou rovnici.

Pro stav 1:

$$p_1 \cdot V_1 = m \cdot r \cdot T_1 \quad (4.11)$$

Pro stav 2:

$$p_2 \cdot V_2 = m \cdot r \cdot T_2 \quad (4.12)$$

### Funkce izotermického děje $it$ :

Základní soustavu rovnic pro řešení izotermického děje tvoří rovnice (4.11) a (4.12). Řešitelnost soustavy a celé funkce je zajištěna po dodržení podmínek řešitelnosti (podkapitola 3.3.1). Co se týče funkce izotermického děje, nepřipustné kombinace neznámých veličin jsou  $p_1$  a  $V_1$ ,  $p_2$  a  $V_2$  nebo  $T$  a  $m$ . Jestliže je neznámou  $T$  a  $m$ , pak je nutné, aby alespoň v jedné z rovnic soustavy byly opět známy hodnoty všech zbývajících proměnných.

Teplota je stavovou veličinou, která je u izotermického děje konstantní. Výstupní vektor teploty  $T$  se tedy určí pomocí funkce *linspace* (300 stejných hodnot). Tlak a objem jsou na sobě závislé stavové veličiny, jeden z výstupních vektorů těchto veličin (v případě funkce  $it$  vektor tlaku  $p$ ) určíme pomocí funkce *linspace*, druhý vektor pomocí cyklu *for* (vektor objemu  $V$ ).

Při volání funkce *it* opět vycházíme z podkapitoly 3.4 a řídíme se dle tab. 4.3.

Tab. 4.3 Vstupní parametry funkce *it*

Pořadí	Veličina (proměnná)	Rozměr (hodnota, vektor)	Jednotka
1.	T	hodnota	[K]
2.	p1	hodnota	[kPa]
3.	V1	hodnota	[m <sup>3</sup> ]
nebo	v1	hodnota	[m <sup>3</sup> /kg]
4.	p2	hodnota	[kPa]
5.	V2	hodnota	[m <sup>3</sup> ]
nebo	v2	hodnota	[m <sup>3</sup> /kg]
6.	plyn=[M; κ]	sloupcový vektor	M [kg/kmol]
			κ [-]
(7.)	m	hodnota	[kg]

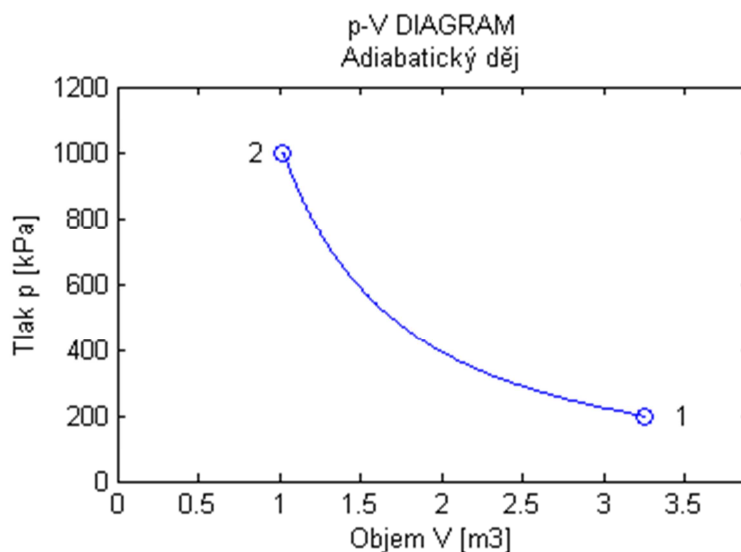
Jako v předchozích případech je zadání vstupního parametru hmotnosti *m* nepovinné, nebude-li zadán, funkce počítá s *m* = 1 kg a s měrnými veličinami. Zadání ostatních vstupních parametrů je povinné (číselná hodnota nebo textový řetězec).

## 4.4 Adiabatický děj

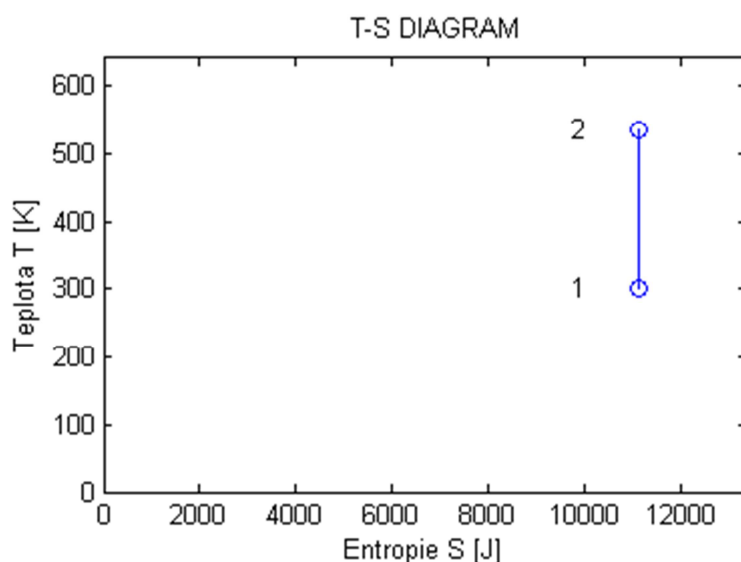
### Základní pojmy a rovnice:

Adiabatický děj, je takový termodynamický děj, který probíhá v izolované termodynamické soustavě, tedy bez výměny tepla s okolím. Tento děj je v *p-V* diagramu znázorněn pomocí adiabaty – viz obr. 4.7. V *T-S* diagramu je pak znázorněn úsečkou rovnoběžnou se svislou osou (osa *T*) – viz obr. 4.8. Základní rovnice adiabatického děje je dána rovnicí adiabaty v *p-V* diagramu (4.13). [6]

$$p \cdot V^{\kappa} = konst. \quad (4.13)$$



Obr. 4.7 *p-V* diagram adiabatického děje



Obr. 4.8 T-S diagram adiabatického děje

### Funkce adiabatického děje *ad*:

Ve výpočtech ve funkci adiabatického děje se využívá znalosti základní rovnice adiabatického děje a dále stavových rovnic pro krajní stavy děje. Princip výpočtů v této funkci je uveden v podkapitole 3.3.2. Určení výstupních vektorů této funkce je rovněž popsáno v podkapitole 3.3.2.

Při volání funkce *ad* vycházíme, jako v případech předchozích funkcí, ze zásad popsaných v podkapitole 3.4 a řídíme se tab. 4.4:

Tab. 4.4 Vstupní parametry funkce *ad*

Pořadí	Veličina (proměnná)	Rozměr (hodnota, vektor)	Jednotka
1.	p1	hodnota	[kPa]
2.	V1	hodnota	[m <sup>3</sup> ]
nebo	v1	hodnota	[m <sup>3</sup> /kg]
3.	T1	hodnota	[K]
4.	p2	hodnota	[kPa]
5.	V2	hodnota	[m <sup>3</sup> ]
nebo	v2	hodnota	[m <sup>3</sup> /kg]
6.	T2	hodnota	[K]
7.	plyn=[M; κ]	sloupcový vektor	M [kg/kmol]
			κ [-]
(8.)	m	hodnota	[kg]

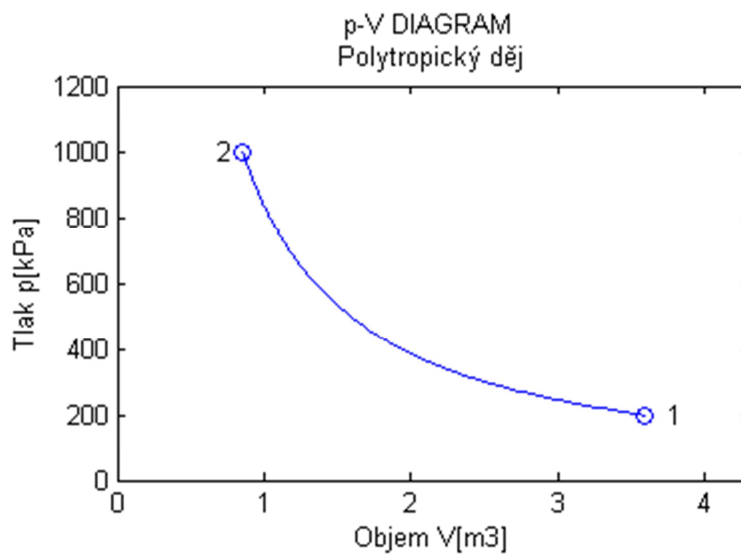
Zadání vstupního parametru hmotnosti  $m$  je nepovinné. Pokud nebude zadán, funkce automaticky uvažuje  $m = 1$  kg a počítá s měrnými veličinami. Všechny další vstupní parametry jsou povinné (musí být zadána číselná hodnota nebo textový řetězec).

## 4.5 Polytropický děj

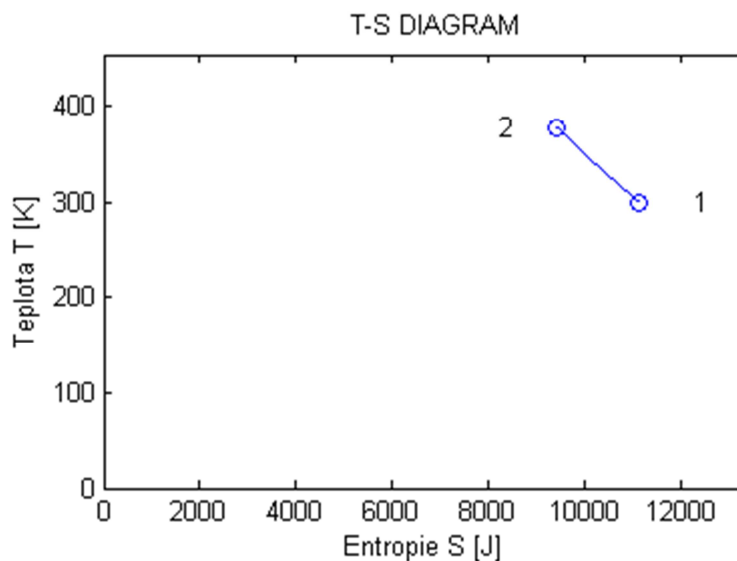
### Základní pojmy a rovnice:

Mezními ději při kompresi a expanzi plynů jsou izotermický děj a adiabatický děj. Tyto děje však nelze v reálných zařízeních uskutečnit, lze se jim pouze přiblížit. Z tohoto důvodu byl zaveden děj, který lépe modeluje změny v reálném světě. Tento děj v  $p$ - $V$  diagramu spadá do oblasti mezi dějem izotermickým a dějem adiabatickým. V  $p$ - $V$  diagramu je znázorněn polytropou – viz obr. 4.9. V  $T$ - $S$  diagramu je tento děj pak vykreslen na obr. 4.10. Základní rovnice polytropického děje je pak dána rovnicí polytropy v tomto diagramu (4.14). [6]

$$p \cdot V^n = konst. \quad (4.14)$$



Obr. 4.9  $p$ - $V$  diagram polytropického děje



Obr. 4.10  $T$ - $S$  diagram polytropického děje

## Funkce polytropického děje *pol*:

Výpočty v této funkci jsou založeny na znalosti základní rovnice polytropického děje pro dva krajní stavy a znalosti stavových rovnic pro tyto dva krajní stavy. Popisy výpočtů a určení výstupních vektorů jsou popsány v podkapitole 3.3.2.

Volání funkce *pol* je založeno na zásadách popsaných v podkapitole 3.4. Dále se také řídíme tab. 4.5.

Tab. 4.5 Vstupní parametry funkce *pol*

Pořadí	Veličina (proměnná)	Rozměr (hodnota, vektor)	Jednotka
1.	$p_1$	hodnota	[kPa]
2.	$V_1$	hodnota	[m <sup>3</sup> ]
nebo	$v_1$	hodnota	[m <sup>3</sup> /kg]
3.	$T_1$	hodnota	[K]
4.	$p_2$	hodnota	[kPa]
5.	$V_2$	hodnota	[m <sup>3</sup> ]
nebo	$v_2$	hodnota	[m <sup>3</sup> /kg]
6.	$T_2$	hodnota	[K]
7.	plyn=[M; $\kappa$ ]	sloupcový vektor	M [kg/kmol] $\kappa$ [-]
8.	$n$	hodnota	[-]
(9.)	$m$	hodnota	[kg]

Jako v předchozích případech je zadání vstupního parametru hmotnosti  $m$  nepovinné. Nebude-li tento vstupní parametr zadán, funkce automaticky uvažuje  $m = 1$  kg a počítá s měrnými veličinami. Ostatní vstupní parametry jsou povinné (musí být zadána číselná hodnota nebo textový řetězec).

## 4.6 Pomocné funkce termodynamických dějů

### 4.6.1 Funkce *apstav*

Při tvorbě funkcí adiabatického a polytropického děje (funkce *ad* a *pol*) se často opakoval jeden druh výpočtu. Proto bylo výhodné zavést pro tento výpočet zvláštní funkci a tuto pak volat v rámci funkcí *ad* a *pol*.

Tato funkce ze znalosti všech stavových veličin jednoho stavu polytropického nebo adiabatického děje, hmotnosti plynu, hodnoty polytropického exponentu nebo v případě adiabatického děje hodnoty Poissonovy konstanty a ze znalosti minimálně jedné hodnoty stavové veličiny ve stavu druhém určí zbývající stavové veličiny v druhém stavu tohoto děje.

Princip výpočtů ve funkci vychází ze znalosti základní rovnice polytropického děje (4.14) a ze stavové rovnice. Přípustné jsou v této funkci maximálně 2 neznámé hodnoty, a to dvě hodnoty stavových veličin, které musí být v jednom stavu, konkrétně ve stavu 2.

Ve funkci je nejprve určen počet neznámých ve stavu 2. Existuje-li pouze jediná neznámá, lze pro určení této neznámé použít funkci *str*. Jsou-li ve funkci dvě neznámé, pak v dalším kroku zjistíme, která proměnná je číselně zadána. Následně s využitím hodnoty této proměnné a pomocí rovnice polytropického (příp. adiabatického) děje určíme jednu z neznámých stavových veličin ve stavu 2. Poslední neznámou veličinu v tomto stavu pak stanovíme s využitím funkce *str*.

### Volání funkce *apstav*:

Funkce *apstav* je na rozdíl od funkcí termodynamických dějů funkcí s pevně daným počtem vstupních parametrů. Při volání musí být zadány hodnoty  $p_1$ ,  $V_1$ ,  $T_1$ , vektor *plyn*,  $n$ ,  $m$  a jedna z hodnot  $p_2$ ,  $V_2$  nebo  $T_2$ . Při zadávání vstupních parametrů je důležité dodržet pořadí zadávaných veličin, rozměr (hodnota, vektor) a jednotku. Řídíme se tabulkou vstupních parametrů této funkce:

Tab. 4.6 Vstupní parametry funkce *apstav*

Pořadí	Veličina (proměnná)	Rozměr (hodnota, vektor)	Jednotka
1.	$p_1$	hodnota	[Pa]
2.	$V_1$	hodnota	[m <sup>3</sup> ]
nebo	$v_1$	hodnota	[m <sup>3</sup> /kg]
3.	$T_1$	hodnota	[K]
4.	$p_2$	hodnota	[Pa]
5.	$V_2$	hodnota	[m <sup>3</sup> ]
nebo	$v_2$	hodnota	[m <sup>3</sup> /kg]
6.	$T_2$	hodnota	[K]
7.	<i>plyn</i> =[M; $\kappa$ ]	sloupcový vektor	M [kg/kmol] $\kappa$ [-]
8.	$n$	hodnota	[-]
9.	$m$	hodnota	[kg]

Důležité je upozornění, že na rozdíl od všech ostatních funkcí (vyjma funkce *str*) je hodnota tlaku zadávána v Pa. Funkce *apstav* je podobně jako funkce *str* nejčastěji volána v jiných funkcích v průběhu výpočtu, ve kterých se tlak vyskytuje v Pa. Tento způsob zadávání tlaku je tedy výhodnější.

Pokud při volání funkce uživatel zná číselnou hodnotu vstupního parametru, pak na místo tohoto parametru dle výše uvedené tabulky zadá tuto číselnou hodnotu v požadované formě a jednotkách, případně název proměnné, která hodnotu obsahuje. Je-li vstupní parametr neznámý, uživatel při volání funkce na její místo zadá libovolný textový řetězec. Z řešitelnosti funkce vyplývá, že textový řetězec smí být použit nejvýše u dvou proměnných z druhého stavu.

Všechny vstupní parametry funkce jsou povinné. Musí být tedy zadána číselná hodnota parametru nebo textový řetězec v případě požadavku na výpočet vstupního parametru.

### 4.6.2 Funkce *stavy* a *stavyd*

V knihovně funkcí jsou děje reprezentovány pomocí vektorů  $p$ ,  $V$ ,  $T$  a  $S$ . Samotné vektory uživatele většinou nezajímají, důležité jsou krajní hodnoty vektorů představující krajní stavy děje. K uložení těchto hodnot do proměnných slouží funkce *stavy*, k jejich vypsání na obrazovku funkce *stavyd*.

#### **Volání funkce *stavy*:**

Funkce má čtyři vstupní parametry – vektory  $p$ ,  $V$ ,  $T$  a  $S$  představující termodynamický děj. Výstupem z funkce hodnoty tlaků  $p_1$ ,  $p_2$ , objemů  $V_1$ ,  $V_2$  (resp. měrných objemů  $v_1$ ,  $v_2$ ), teplot  $T_1$ ,  $T_2$  a entropií  $S_1$ ,  $S_2$  (resp. měrných entropií  $s_1$ ,  $s_2$ ) v krajních stavech tohoto děje v pořadí, v jakém jsou zde vypsány.

#### **Volání funkce *stavyd*:**

Vstupní parametry této funkce jsou stejné jako u funkce *stavy*. Výstupní parametry funkce nemá. Provede pouze výpis hodnot stavových veličin v krajních stavech děje na obrazovku. Slouží pro rychlou orientaci ve výpočtu.

## 5 ENERGETICKÉ VELIČINY TERMODYNAMICKÝCH DĚJŮ

V průběhu termodynamických dějů dochází k energetickým změnám, které jsou charakterizovány velikostí objemové práce, technické práce a tepla, popřípadě měrné objemové práce, měrné technické práce a měrného tepla. Ze znalosti těchto veličin pak můžeme určit energetické pochody mezi soustavou a okolím.

Tyto veličiny vyjadřují působení soustavy na své okolí a závisí na podmínkách, při kterých ke změnám v soustavě dochází. Jejich velikost závisí na termodynamickém ději (na křivce mezi počátečním a konečným stavem). Tyto veličiny nejsou stavové a většinou je lze v grafech vyjádřit prostřednictvím plochy.

### 5.1 Objemová práce

#### Základní pojmy a veličiny:

Objemová práce je taková práce, kterou soustava vykoná nebo spotřebuje během změny svého objemu. Pokud uvažujeme plyn ve válci spolu s pohyblivým pístem, je tato práce dána působením síly  $F$  působící na píst po dráze pístu  $l$ .

Velikost objemové práce je dána plochou pod křivkou daného děje (v našem případě děj 12) v  $p$ - $V$  diagramu. Totéž platí pro měrnou objemovou práci ovšem v  $p$ - $v$  diagramu.

Objemová práce  $A_{o12}$  a měrná objemová práce  $a_{o12}$  jsou dány vztahy:

$$A_{o12} = \int_1^2 p \cdot dV \quad [\text{J}] \quad (5.1)$$

$$a_{o12} = \int_1^2 p \cdot dv \quad [\text{J} \cdot \text{kg}^{-1}] \quad (5.2)$$

kde  $p$  je tlak plynu,  $dV$  element objemu,  $dv$  element měrného objemu plynu.

Pokud dojde k expanzi plynu, je objemová práce práci vykonanou a je kladná. Naopak dojde-li ke kompresi, je spotřebována práce dodaná z okolí a objemová práce je záporná.

Mezi objemovou prací a měrnou objemovou prací platí vztah:

$$A_{o12} = m \cdot a_{o12} \quad (5.3)$$

kde  $m$  je hmotnost pracovní látky. [6]

#### Funkce pro výpočet objemové práce:

Funkce s názvem *praceo* umožní uživateli určit hodnotu objemové práce  $A_o$  nebo měrné objemové práce  $a_o$  termodynamického děje reprezentovaného vektory  $p$  a  $V$  (resp.  $v$ ).

## Princip výpočtů ve funkci:

Výpočet hodnoty objemové práce provedeme numerickým integrováním pomocí obdélníkové formule. Princip spočívá v sečtení obsahů 299 malých obdélníků o výšce  $\frac{p_k + p_{k+1}}{2}$  a o šířce  $V_{k+1} - V_k$ , kde  $k \in \langle 1; 299 \rangle$ , což můžeme matematicky zapsat následovně:

$$A_o = \sum_{k=1}^{299} \frac{p_k + p_{k+1}}{2} \cdot (V_{k+1} - V_k) \quad (5.4)$$

Způsob výpočtu objemové práce a měrné objemové práce je stejný. Zda funkce počítá objemovou nebo měrnou objemovou práci, záleží pouze na tom, jestli je vstupním parametrem objem  $V$  nebo měrný objem  $v$ . V případě zadaného objemu funkce počítá objemovou práci, je-li zadán měrný objem, funkce počítá měrnou objemovou práci.

## Volání funkce:

Vstupní parametry funkce *praceo* tvoří řádkový vektor tlaku  $p$  [kPa] a řádkový vektor objemu  $V$  [m<sup>3</sup>], popřípadě měrného objemu  $v$  [m<sup>3</sup>·kg<sup>-1</sup>]. Každý z těchto vektorů obsahuje 300 hodnot. Vektory odpovídají termodynamickému ději, jehož objemovou nebo měrnou objemovou práci funkce určuje. Tyto vektory může uživatel získat použitím odpovídající funkce termodynamického děje (např. *ib*, *it*, *ich*).

Výstupním parametrem je hodnota objemové práce, popřípadě měrné objemové práce termodynamického děje.

V termodynamických výpočtech opatřujeme objemovou práci indexy děje, pro který tuto práci počítáme, např. počítáme-li objemovou práci pro děj mezi stavy 2 a 3, označíme ji  $A_{o23}$ . Funkce *praceo* je užívána pro výpočet práce (objemové či měrné objemové) obecného děje. Indexování proměnné pro hodnotu práce přímo v této funkci by bylo zcestné, neboť uživatel může počítat objemovou práci libovolného děje svého konkrétního výpočtu a my předem nevíme, o jak označený děj půjde. Indexování objemové práce může provést uživatel sám při volání této funkce. Proměnnou, do které bude uložena výsledná hodnota, doporučujeme uživateli označit podle toho, zda počítá objemovou práci nebo měrnou objemovou práci, a sice velkým nebo malým písmenem „A“ spolu s písmenem „o“ k rozlišení objemové práce od technické, a poté opatřit tuto proměnnou čísly děje, pro který práci počítá.

## 5.2 Technická práce

### Základní pojmy a veličiny:

Technická práce, někdy nazývaná jako tlaková, je práce, kterou získáváme na výstupních hřídelích rotačních strojů. Koná se pouze v případě, mění-li se tlak plynu.

Velikost technické práce se určuje obdobným způsobem jako velikost objemové práce. Je rovněž dána plochou pod křivkou daného děje mezi stavy 1 a 2 v  $p$ - $V$  diagramu, avšak v tomto případě jde o plochu směřující od křivky diagramu ke svislé ose  $p$  (při výpočtu objemové práce plocha směřuje od křivky k ose  $V$ ). Totéž platí pro měrnou technickou práci v  $p$ - $v$  diagramu.

Velikost technické práce  $A_{t12}$  a měrné technické práce  $a_{t12}$  je definována následovně:

$$A_{t12} = - \int_1^2 V \cdot dp \quad (5.5)$$

$$a_{t12} = - \int_1^2 v \cdot dp \quad (5.6)$$

kde  $V$  je objem plynu,  $v$  měrný objem plynu a  $dp$  element tlaku.

Ve vztazích (5.5) a (5.6) se před určitým integrálem objevuje znaménko mínus. Je tomu tak kvůli zavedené znaménkové konvenci, která nám říká, že technická práce je kladná právě tehdy, když tlak plynu během termodynamického děje 12 klesá. Roste-li tlak plynu během děje 12, technická práce se spotřebovává a je záporná.

Mezi technickou a měrnou technickou prací platí vztah:

$$A_{t12} = m \cdot a_{t12} \quad (5.7)$$

kde  $m$  je hmotnost pracovní látky. [6]

### Funkce pro výpočet technické práce:

Hodnotu technické práce  $A_t$ , případně hodnotu měrné technické práce  $a_t$  určí uživatel pomocí funkce s názvem *pracet*, a sice na základě znalosti vektorů  $p$  a  $V$  (resp.  $v$ ), které reprezentují konkrétní termodynamický děj.

### Princip výpočtů ve funkci:

Výpočet velikosti technické práce provedeme opět numerickým integrováním. Sečteme obsahy 299 malých obdélníků, jejichž výška je  $\frac{V_k + V_{k+1}}{2}$  a šířka  $p_{300-k+1} - p_{300-k}$ , kde  $k \in \langle 1; 299 \rangle$ . Vztah pro výpočet technické práce pak matematicky zapíšeme takto:

$$A_t = - \sum_{k=1}^{299} \frac{V_k + V_{k+1}}{2} \cdot (p_{300-k+1} - p_{300-k}) \quad (5.8)$$

Určení velikosti technické a měrné technické práce je založeno na stejném principu. Zda funkce počítá technickou nebo o měrnou technickou práci záleží pouze na tom, jestli uživatel jako vstupní parametr funkce zadá objem  $V$  nebo měrný objem  $v$ .

### Volání funkce:

Volání funkce *pracet* je podobné jako volání funkce *praceo*. Funkce *pracet* má dva vstupní parametry – řádkový vektor tlaku  $p$  a řádkový vektor objemu  $V$ , eventuálně měrného objemu  $v$ . Oba tyto vektory obsahují 300 hodnot a určují termodynamický

děj, jehož technickou práci počítáme. Vektory lze získat užitím odpovídající funkce termodynamického děje.

Výstupním parametrem funkce je hodnota technické práce, resp. měrné technické práce, uživatelem daného termodynamického děje.

Pro indexování technické práce v termodynamických výpočtech platí stejná zákonitost jako je tomu v případě objemové práce (podkapitola 5.1). Proměnnou, do které je ukládána výsledná hodnota technické práce, se pro přehlednost výpočtů uživateli doporučuje pojmenovat podle toho, zda počítá technickou nebo měrnou technickou práci, a to velkým nebo malým písmenem „A“ spolu s písmenem „t“ k rozlišení technické práce od objemové. Dále je výhodné proměnnou označit čísly děje, pro něž uživatel práci určuje.

## 5.3 Teplo

### Základní pojmy a veličiny:

Teplo představuje formu přenosu tepelné energie mezi termodynamickou soustavou a okolím při současné změně stavu soustavy. Je to tedy část vnitřní energie soustavy, kterou soustava vymění s jinou soustavou (např. s okolním prostředím).

Velikost tepla, které předá soustava s pracovní látkou během termodynamického děje jiné soustavě, je dána plochou pod křivkou daného děje (děj 12) v  $T$ - $S$  diagramu. Totéž platí pro měrné teplo, avšak v  $T$ - $s$  diagramu.

Velikost tepla  $Q_{12}$ , případně měrného tepla  $q_{12}$ , se určuje následujícím způsobem:

$$Q_{12} = \int_1^2 T \cdot dS \quad (5.9)$$

$$q_{12} = \int_1^2 T \cdot ds \quad (5.10)$$

kde  $T$  je teplota plynu,  $dS$  element entropie a  $ds$  element měrné entropie.

I v případě tepla platí znaménková konvence. Kladné znaménko přiřadíme teplu přivedenému do soustavy, záporné znaménko pak teplu odvedenému ze soustavy.

Mezi teplem a měrným teplem platí vztah:

$$Q_{12} = m \cdot q_{12} \quad (5.11)$$

kde  $m$  je hmotnost pracovní látky. [6]

### Funkce pro výpočet tepla:

Pro výpočet tepla z vektorů teploty  $T$  a entropie  $S$  daného termodynamického děje byla v knihovně funkcí termodynamických výpočtů sestavena funkce s názvem *teplo*.

### Princip výpočtů ve funkci:

Výpočet velikosti tepla ve funkci provedeme numericky pomocí obdélníkové formule a to tak, že sečteme obsahy 299 malých obdélníků, jejichž výška je  $\frac{T_k + T_{k+1}}{2}$  a šířka  $S_{k+1} - S_k$ , kde  $k \in \{1; 299\}$ . Tuto operaci lze zapsat následovně:

$$Q = \sum_{k=1}^{299} \frac{T_k + T_{k+1}}{2} \cdot (S_{k+1} - S_k) \quad (5.12)$$

Princip výpočtu tepla a měrného tepla je stejný. Teplo  $Q$  tato funkce určuje v případě, že je jako vstupní parametr zadána entropie  $S$ , měrné teplo  $q$  pak funkce stanoví v případě zadání měrné entropie  $s$ .

### Volání funkce:

Tato funkce má dva vstupní parametry – řádkový vektor teploty  $T$  a řádkový vektor entropie  $S$  (resp. měrné entropie  $s$ ). Tyto vektory obsahují každý 300 hodnot a určují termodynamický děj, jehož teplo počítáme. Vektory je možno získat užitím odpovídající funkce termodynamického děje.

Výstupním parametrem funkce je hodnota tepla děje, případně měrného tepla děje, který zastupují vektory vstupující do funkce.

Pro indexování tepla platí stejná pravidla jako v případě předchozích energetických veličin (podkapitoly 5.1 a 5.2).

## 6 VYKRESLENÍ GRAFŮ

Termodynamické děje a cykly je pro lepší názornost vhodné graficky zobrazit. V termodynamice jsou nejčastěji používány grafické závislosti objemu (příp. měrného objemu) na tlaku a entropie (příp. měrné entropie) na teplotě. Jsou to tedy  $p$ - $V$  (resp.  $p$ - $v$ ) diagramy a  $T$ - $S$  (resp.  $T$ - $s$ ) diagramy.

V knihovně funkcí jsou termodynamické cykly reprezentovány prostřednictvím matic tlaku  $p$ , objemu  $V$  (měrného objemu  $v$ ), teploty  $T$  a entropie  $S$  (měrné entropie  $s$ ). Matice jsou ve funkcích označovány jako  $mat\_p$ ,  $mat\_V$  ( $mat\_v$ ),  $mat\_T$  a  $mat\_S$  ( $mat\_s$ ). Platí, že každý řádek matice, představuje jeden termodynamický děj cyklu. Každá matice má tedy 300 sloupců a počet řádků matice odpovídá počtu dějů cyklu. Řádky matice musí být uspořádány tak, aby na sebe jednotlivé děje navazovali. V případě, že na sebe nebudou řádky v matici navazovat, dojde při vykreslení grafů k nepřesnému popisu krajních stavů cyklu.

### 6.1 Vykreslení $p$ - $V$ diagramu

K vykreslení  $p$ - $V$  ( $p$ - $v$ ) diagramu termodynamického děje nebo cyklu slouží v knihovně funkcí funkce s názvem *pvdiag*.

#### Popis principů ve funkci:

Vstupními parametry funkce jsou matice tlaku a matice objemu (příp. měrného objemu). V prvním kroku zjistíme, zda byla zadána matice objemu nebo matice měrného objemu. Tato informace je důležitá k popisu os malým nebo velkým písmenem „ $V$ “. V následujícím kroku zjistíme počet řádků matice, čímž určíme počet dějů cyklu. Poté funkce postupně provede vykreslení všech dějů cyklu. Vykreslením děje cyklu se rozumí vykreslení křivky děje a zvýraznění krajních stavů děje kolečkem pomocí příkazu *plot* a také popis krajních stavů děje stavu pomocí příkazu *text*. Dále pak popis osy  $x$  a  $y$  pomocí příkazů *xlabel* a *ylabel* a samotného grafu pomocí příkazu *title*. Nakonec je třeba nastavit rozsah os, k tomu slouží příkaz *axis*. Funkce všech příkazů popsanych v tomto odstavci jsou k dispozici v tab. 1.5.

Pokud je zadána matice o jednom řádku, tedy vektor, znamená to, že se vykreslí pouze děj. V tomto případě je na místě popsat graf také údajem druhu termodynamického děje.

#### Volání funkce:

Vstupními parametry této funkce jsou matice tlaku a objemu (resp. měrného objemu) reprezentující termodynamický cyklus, případně vektory tlaku a objemu (resp. měrného objemu) reprezentující termodynamický děj. Podoba matic je popsána v úvodní části této kapitoly. Dalším vstupním parametrem je sloupcový vektor *plyn*, který na prvním řádku obsahuje molární hmotnost plynu a na druhém hodnotu Poissonovy konstanty.

Výstupní parametry funkce nemá, provede se pouze zobrazení  $p$ - $V$  (resp.  $p$ - $v$ ) diagramu.

## 6.2 Vykreslení $T$ - $S$ diagramu

V knihovně funkcí k vykreslení  $T$ - $S$  (příp.  $T$ - $s$ ) diagramu termodynamického děje či cyklu slouží funkce pojmenovaná jako *tsdiag*.

### Popis principů ve funkci:

Vstupními parametry této funkce jsou matice teploty a entropie (měrné entropie). Ve funkci v první řadě zjistíme, zda byla zadána matice entropie nebo měrné entropie, a na této informaci pak založíme popis vodorovné osy. Dále zjistíme počet řádků matice a provedeme vykreslení všech dějů cyklu spolu s popisem diagramu. Vykreslení provedeme stejným způsobem jako v případě  $p$ - $V$  diagramu (popsáno v podkapitole 6.1.).

### Volání funkce:

Jak je již psáno výše, vstupními parametry funkce *tsdiag* jsou matice teploty a entropie reprezentující termodynamický cyklus, jejichž podoba je popsána v úvodní části kapitoly, nebo případně vektory teploty a entropie představující termodynamický děj. Dalším vstupním parametrem funkce je sloupcový vektor *plyn*, jehož podoba je popsána rovněž v předchozí podkapitole.

Výstupní parametry této funkce rovněž nemá, provede se pouze vyobrazení  $T$ - $S$  ( $T$ - $s$ ) diagramu.

## 7 APLIKACE KNIHOVNY FUNKCÍ

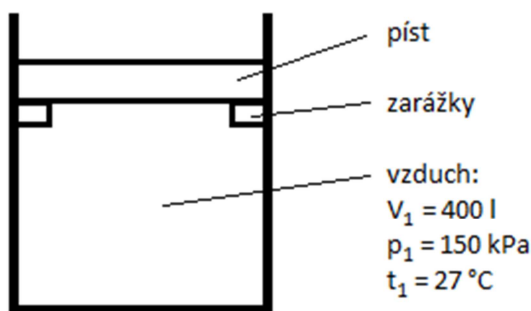
V této kapitole si pro přiblížení práce s knihovnou funkcí termodynamických dějů v MATLABu ukážeme její použití na konkrétním příkladu ze sbírek [7]. Dále jsou zde pro uživatele uvedeny některé užitečné tipy a příkazy MATLABu, které jsou určeny k usnadnění práce s knihovnou.

### 7.1 Výpočet příkladu s využitím knihovny funkcí

Pro tento výpočet byl vybrán příklad 1.6 ze sbírky [7], který jsme modifikovali pro ideální plyn.

Ve válci s pístem je na počátku vzduch o tlaku 150 kPa a teplotě 27 °C. Poloha pístu je dána zářkami podle obrázku obr. 7.1 a uzavřený objem je 400 l. Hmotnost pístu je taková, že tlak k jeho pohybu vzhůru musí dosáhnout hodnoty 350 kPa. Vzduch je ohříván tak, že se jeho objem zvětší 1,5krát. Určete:

- konečnou teplotu
- objemovou práci vykonanou vzduchem
- celkové přivedené teplo.



Obr. 7.1 Válec s volným pístem

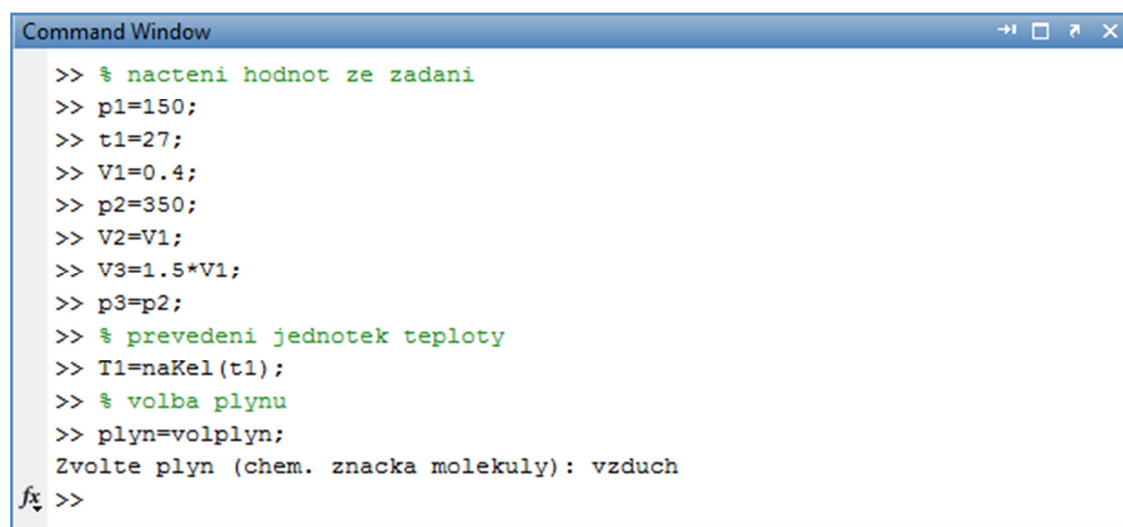
Vzduch se vzhledem k nízkému tlaku a teplotě chová jako ideální plyn. Na začátku výpočtu do MATLABu vložíme hodnoty proměnných známých ze zadání. Při vkládání je nutno brát ohled na jednotky. K převodu jednotek teploty na Kelviny využijeme funkce *naKel*. Dále provedeme pomocí funkce *volplyn* volbu plynu. Všechny tyto operace můžeme vidět na obr. 7.2.

Změny, kterými projde vzduch během děje popsaného v zadání, lze popsat dvěma základními ději. Izochorický děj – děj 12 probíhající do doby, než začne zvedání pístu, izobarický děj – děj 23 probíhající v průběhu zvedání pístu.

Pomocí funkce *ich* určíme ze znalosti stavových veličin v krajních stavech děje 12 teplotu  $T_2$ . Funkce *ich* vytvoří vektory  $p_{12}$ ,  $V_{12}$ ,  $T_{12}$  a  $S_{12}$ , které reprezentují děj 12. Důležité je, aby výstupní proměnné měly jiný název než proměnné, které již existují. Jinak by došlo k přepsání hodnot v existujících proměnných. Jedinou neznámou je  $T_2$ , avšak musíme zadat také požadavek na výpočet hmotnosti plynu  $m$  a to z toho důvodu, že uvažujeme objem  $V$  nikoliv měrný objem  $v$  a pokud bychom požadavek na výpočet hmotnosti nezadali, funkce by automaticky uvažovala výpočet s měrnými veličinami.

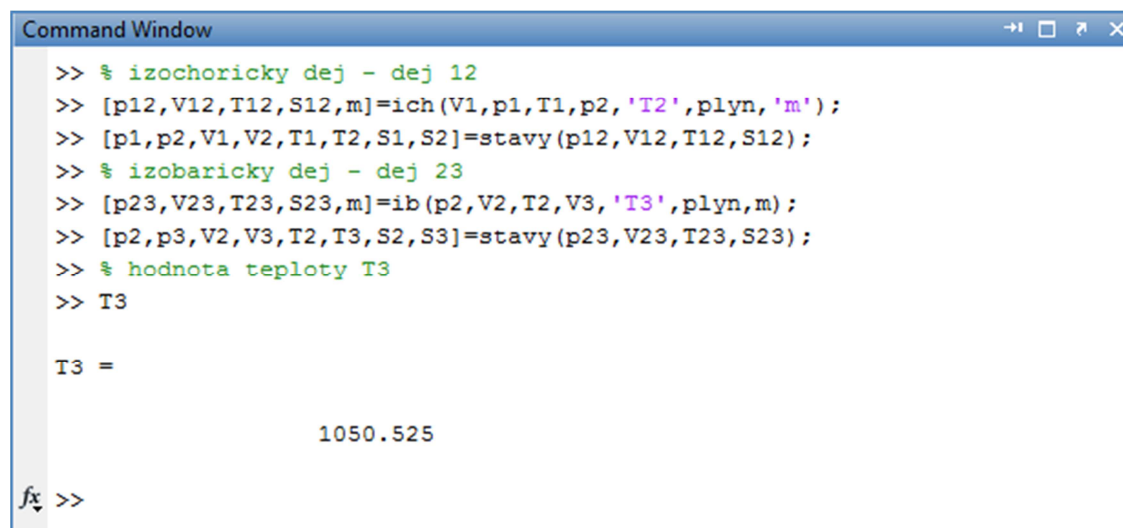
Po vytvoření vektorů  $p_{12}$ ,  $V_{12}$ ,  $T_{12}$  a  $S_{12}$  použijeme funkci *stavy*, prostřednictvím níž budou do proměnných uloženy hodnoty stavových veličin v krajních stavech. V tomto okamžiku známe všechny stavové veličiny ve stavech 1 a 2.

V následujícím kroku výpočtu použijeme funkci *ib* pro děj 23. Funkce ze známých stavových veličin v krajních stavech tohoto děje určí výstupní vektory  $p_{23}$ ,  $V_{23}$ ,  $T_{23}$  a  $S_{23}$ . Dále opět prostřednictvím funkce *stavy* uložíme do proměnných hodnoty stavových veličin v krajních stavech tohoto děje. Nakonec zobrazíme proměnnou  $T_3$ , což je námi hledaná konečná teplota  $T_3$ . Výpočet popsany v tomto a v předchozím odstavci lze vidět na obr. 7.3. Podle výpočtu je konečná teplota  $T_3 = 1050,525$  K.



```
Command Window
>> % nacteni hodnot ze zadani
>> p1=150;
>> t1=27;
>> V1=0.4;
>> p2=350;
>> V2=V1;
>> V3=1.5*V1;
>> p3=p2;
>> % prevedeni jednotek teploty
>> T1=naKel(t1);
>> % volba plynu
>> plyn=volplyn;
Zvolte plyn (chem. znacka molekuly): vzduch
fx >>
```

Obr. 7.2 Vložení údajů ze zadání



```
Command Window
>> % izochoricky dej - dej 12
>> [p12,V12,T12,S12,m]=ich(V1,p1,T1,p2,'T2',plyn,'m');
>> [p1,p2,V1,V2,T1,T2,S1,S2]=stavy(p12,V12,T12,S12);
>> % izobaricky dej - dej 23
>> [p23,V23,T23,S23,m]=ib(p2,V2,T2,V3,'T3',plyn,m);
>> [p2,p3,V2,V3,T2,T3,S2,S3]=stavy(p23,V23,T23,S23);
>> % hodnota teploty T3
>> T3

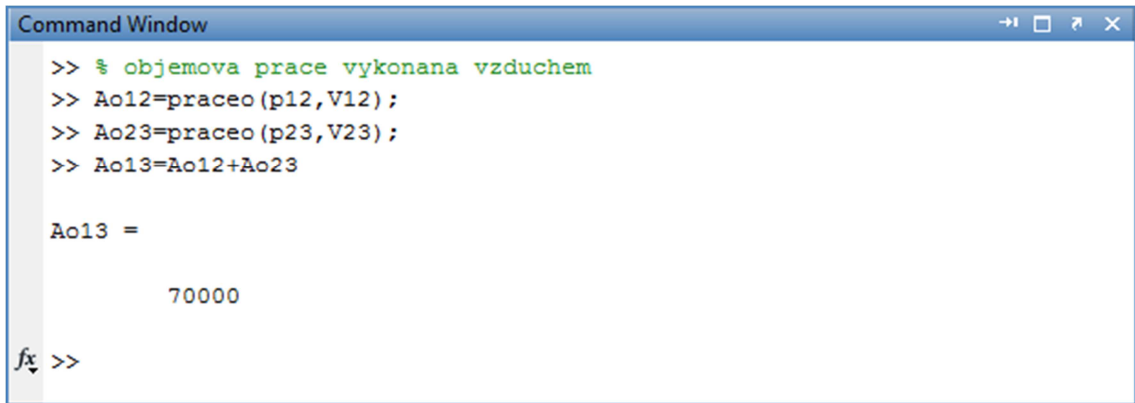
T3 =

          1050.525

fx >>
```

Obr. 7.3 Stanovení konečné teploty  $T_3$

Celkovou objemovou práci vypočítáme tak, že sečteme objemové práce jednotlivých dějů. Jelikož je objemová práce izochorického děje nulová, stačilo by určit pouze objemovou práci děje izobarického. Pro větší názornost ale provedeme výpočet celkové práce ze součtu prací obou dějů. Vstupními parametry funkce pro výpočet objemové práce děje jsou, jak můžeme vidět na obr. 7.4, vektory tlaku a objemu daného děje. Výsledkem výpočtu je celková objemová práce  $A_{o13} = 140 \text{ kJ}$ .



```
Command Window
>> % objemova prace vykonana vzduchem
>> Ao12=praceo(p12,V12);
>> Ao23=praceo(p23,V23);
>> Ao13=Ao12+Ao23

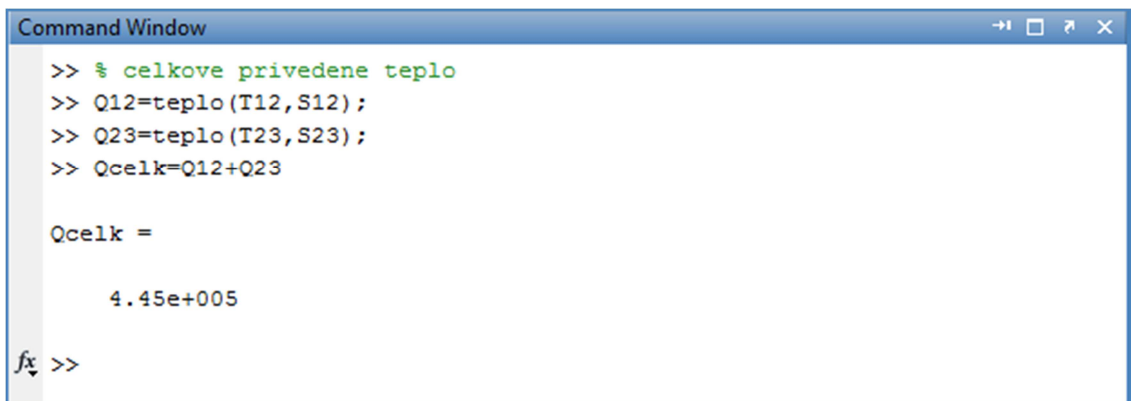
Ao13 =

      70000

fx >>
```

Obr. 7.4 Stanovení celkové objemové práce  $A_{o13}$

Nakonec vypočteme celkové přivedené teplo. Toto teplo stanovíme součtem tepla děje 12 a tepla děje 23. Vstupními parametry funkcí pro výpočet tepla děje jsou vektory teploty a entropie reprezentující daný děj. Výpočet celkového tepla je možno vidět na obr. 7.5. Celkové přivedené teplo je  $Q_{celk} = 445 \text{ kJ}$ . Tímto jsme splnily všechny požadavky zadání.



```
Command Window
>> % celkove privedene teplo
>> Q12=teplo(T12,S12);
>> Q23=teplo(T23,S23);
>> Qcelk=Q12+Q23

Qcelk =

  4.45e+005

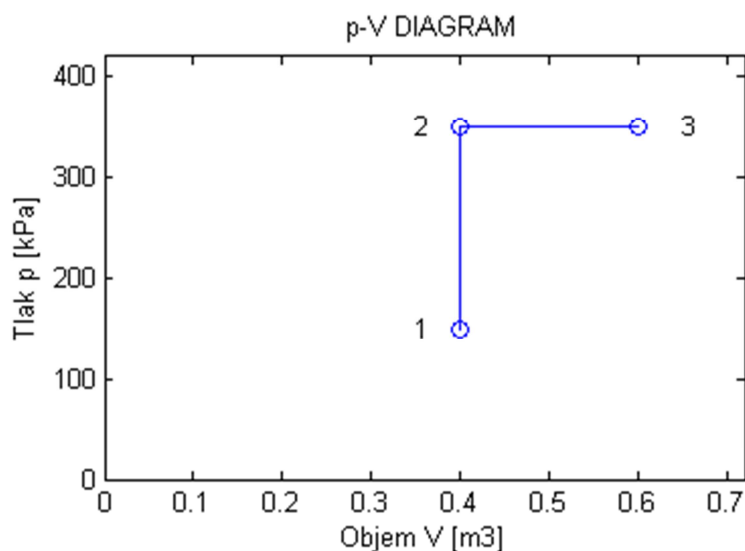
fx >>
```

Obr. 7.5 Stanovení celkového přivedeného tepla  $Q_{celk}$

Závěrem si pro přiblížení použití funkce pro zobrazení  $p$ - $V$  diagramu *pvdiag* vykreslíme do jednoho obrázku  $p$ - $V$  diagram dějů 12 a 23. Volání funkce *pvdiag* je spolu s vytvořením vstupních matic této funkce zobrazeno na obr.7.6. Samotný  $p$ - $V$  diagram pak na obr. 7.7.

```
Command Window
>> % p-V diagram
>> mat_p(1,:)=p12;
>> mat_p(2,:)=p23;
>> mat_V(1,:)=V12;
>> mat_V(2,:)=V23;
>> pvdiag(mat_p,mat_V,plyn);
OBJEM V nebo MĚRNÝ OBJEM v? [zadejte textový řetězec "V" nebo "v"]: 'V'
fx >>
```

Obr. 7.6 Příkazy vedoucí k vykreslení  $p$ - $V$  diagramu



Obr. 7.7 Vykreslení  $p$ - $V$  diagramu

## 7.2 Užitečné tipy pro uživatele

### Načtení knihovny funkcí:

Pro správnou funkčnost knihovny je třeba, aby byly všechny její funkce načteny v jedné složce. Po spuštění programu MATLAB před začátkem samotné práce s knihovnou pak v poli označeném jako Current Folder tuto složku nejprve vyhledáme a zvolíme její použití.

### Nápověda k funkcím:

V úvodu zdrojového textu každé funkce jsou uvedeny základní informace týkající se dané funkce. Patří mezi ně informace o tom, jaký výpočet funkce provádí (k čemu slouží), informace o vstupních parametrech (pořadí, jednotky, počet neznámých atd.), případně jakým způsobem se dají získat vstupní parametry funkce nebo kde lze funkci použít. Zobrazení této nápovědy program MATLAB provede po zadání příkazu „*help* nazev\_funkce“ do okna Command Window.

### Užitečné příkazy a povely:

Aby byla práce v programu MATLAB pro uživatele co nejpříjemnější, je k usnadnění a zpřehlednění výpočtů výhodné znát několik základních příkazů, které se týkají práce s proměnnými, s funkcemi a číselného formátu výstupních hodnot. Tyto příkazy spolu s jejich použitím jsou uvedeny v tab. 7.1. Další příkazy je možné najít např. ve zdroji [3] nebo [4].

Tab. 7.1 Užitečné příkazy [4]

Příkaz	Příklad použití	Význam příkazu
clc	clc	vyčistí hlavní okno (Command Window), zadaná čísla a obsahy proměnných zůstávají nezměněny
clear	clear all	vymazání všech proměnných z paměti
	clear A	vymazání proměnné A z paměti
help	help sin	zobrazí nápovědu k funkci (v tomto případě k funkci sin)
	help elfun	zobrazí přehled elementárních funkcí
	help ops	zobrazí přehled operátorů a speciálních znaků
;	A=12*25+4;	středník slouží k potlačení výpisu výsledku na obrazovku (výpočet se provede, ale není zobrazen); je možno jej použít u výrazů, které se potvrzují tlačítkem ENTER
<b>Číselné formáty:</b>		
short	format short	přednastavený formát zobrazení čísel - 5 platných číslic, 4 desetinná místa, pevná řádová čárka
long	format long	15 platných číslic, 14 desetinných míst, pevná řádová čárka
short e	format short e	5 platných číslic, 4 desetinná místa, plovoucí řádová čárka
long e	format long e	15 platných číslic, 14 desetinných míst, plovoucí řádová čárka
short g	format short g	automaticky vybírá nejvhodnější formát mezi short a short e
long g	format long g	automaticky vybírá nejvhodnější formát mezi long a long e

## ZÁVĚR

Knihovna funkcí vytvořená v rámci bakalářské práce usnadňuje provedení základních termodynamických výpočtů s ideálními plyny. Funkce knihovny byly sestaveny tak, aby byla práce s nimi pro uživatele co nejpohodlnější a uživatel rychle a snadno pochopil jejich použití.

V technické praxi, například při návrhovém výpočtu tepelných pochodů v pístových motorech, kompresorech a jiných zařízeních pracujících na základě termodynamických změn plynů, se konstruktér, případně projektant, setkává s neustále se opakujícím cyklem výpočtů. Prostřednictvím těchto opakujících se výpočtů pak hledá ideální stav vstupujících veličin pro svoji konkrétní aplikaci. Tato část návrhu může být v určitých případech velmi zdlouhavá, zvláště pak, nemá-li konstruktér k dispozici program, který by výpočty prováděl automaticky. Zejména k urychlení této fáze návrhu byla tato knihovna funkcí vytvořena.

Pokud má konstruktér či projektant k dispozici naprogramované své rutinní výpočty ve funkcích, do kterých pouze zadá vstupní parametry, a získá z nich okamžitě nebo po několika jednoduchých operacích výsledky, uspoří si tím určitý čas, který pak může věnovat důležitějším činnostem. Tím roste efektivita jeho práce. Z toho důvodu je programování nejrozličnějších výpočtů, včetně těch termodynamických, a vytváření aplikací pro operace, které by jinak konstruktér prováděl každodenně, hospodárným přínosem pro firmy, případně pro konstruktéra samotného.

Nutno však podotknout, že v této práci se zabýváme pouze ideálními plyny, jejich směsmi a tedy také ideálními změnami těchto plynů. V případě, že uživatel použije tuto knihovnu funkcí pro reálné aplikace, je třeba zvážit vhodnost jejího použití a případně výsledky uvažovat pouze jako orientační.

Dále je tato knihovna funkcí určena pro studenty, kterým by měla sloužit k ověření svých výpočtů a být tak užitečnou pomůckou zejména v základním kurzu termomechaniky, případně přiblížit práci s programem MATLAB.

Knihovna funkcí základních termodynamických výpočtů je zpracována tak, že každou funkci lze použít po zadání příkazu k jejímu volání a zapsání vstupních parametrů po spuštění MATLABu v okně Command Window. Takový způsob zadávání může být pro uživatele, který není zvyklý v tomto programu pracovat, nepohodlný a mohl by jej od používání knihovny funkcí odradit. Pohodlnější a tudíž i uživatelsky příjemnější by bylo, kdyby se po zavolání funkce prostřednictvím zapsání jejího názvu nebo případně jiným způsobem na obrazovce objevilo zvláštní okno funkce, kde by se do vyznačených polí vkládaly příslušné veličiny. Program MATLAB podobné způsoby nabízí. Blíže jsou tyto způsoby popsány ve zdroji [3], avšak z důvodů časového presu, se mi je nepodařilo blíže prozkoumat. Proto byl zvolen způsob zadávání prostřednictvím příkazu k volání funkce v okně Command Window.

Během vytváření funkcí v rámci bakalářské práce jsem si rozšířil znalosti programovacích metod v programu MATLAB získané v rámci předmětu informatika. Podařilo se mi načerpat nové zkušenosti s tímto programem. Přesvědčil jsem se o tom, že vytváření programů či funkcí, není vyjma těch nejjednodušších nikdy přímočaré. Ba naopak cesta k vytvoření kýžené funkce provádějící požadované úkony a vracející správné výsledky je často velmi spletitá.

## SEZNAM POUŽITÝCH ZDROJŮ

- [1] MATLAB. *The Language of Technical Computing* [online]. 2011 [cit. 2012-04-25]. Dostupné z WWW:  
[http://www.mathworks.com/tagteam/70533\\_91199v01\\_MATLABDataSheet\\_v9.pdf?s\\_cid=ML2012\\_bb\\_datasheet](http://www.mathworks.com/tagteam/70533_91199v01_MATLABDataSheet_v9.pdf?s_cid=ML2012_bb_datasheet)
- [2] CHAPMAN, Stephen J. *Essential of MATLAB Programming*. Stamford: Cengage Learning, 2006. ISBN 0-495-29568-X.
- [3] ZAPLATÍLEK, Karel a Bohuslav DOŇAR. *MATLAB: tvorba uživatelských aplikací*. 1. vyd. Praha: BEN, 2004, 215 s. ISBN 80-730-0133-0.
- [4] ZAPLATÍLEK, Karel a Bohuslav DOŇAR. *MATLAB: pro začátečníky*. 2. vyd. Praha: BEN - technická literatura, 2005, 151 s. ISBN 80-730-0175-6.
- [5] BORGNACKE, C, Richard Edwin SONNTAG, Gordon J VAN WYLEN a Richard Edwin SONNTAG. *Fundamentals of thermodynamics*. 7th ed. Hoboken, NJ: Wiley, c2009, xvii, 894 p. ISBN 978-047-0041-925.
- [6] PAVELEK, Milan. *Termomechanika*. Vyd. 3. přeprac. /. Brno: Akademické nakladatelství CERM, 2003, 284 s. ISBN 80-214-2409-5.
- [7] VACEK, Václav a Jiří NOŽIČKA. *Příručka z termodynamiky s příklady: [určeno pro posl. fak. strojní ČVUT v Praze]*. 1. vyd. Praha: ČVUT, 1993, 113 s. ISBN 80-010-1008-2.
- [8] BALMER, Robert T. *Modern engineering thermodynamics*. Vyd. 3. přeprac. Boston: Academic Press, 2011, xxiii, 801 p. ISBN 978-012-3749-963.
- [9] MASSOUD, Mahmoud. *Engineering thermofluids: thermodynamics, fluid mechanics, and heat transfer*. Berlin: Springer, 2005, xxiv, 1119 p. ISBN 35-402-2292-8.
- [10] PALM, William J. *Introduction to MATLAB for engineers*. 3rd ed. New York: McGraw-Hill, c2011, xii, 564 p. ISBN 00-735-3487-0.

## SEZNAM POUŽITÝCH VELIČIN

Symbol	Jednotka	Název veličiny
$A_o$	J	objemová práce
$a_o$	$\text{J}\cdot\text{kg}^{-1}$	měrná objemová práce
$A_t$	J	technická práce
$a_t$	$\text{J}\cdot\text{kg}^{-1}$	měrná technická práce
$A_r$	-	relativní atomová hmotnost
$c_p$	$\text{J}\cdot\text{kg}^{-1}\cdot\text{K}^{-1}$	měrná tepelná kapacita za konstantního tlaku
$c_v$	$\text{J}\cdot\text{kg}^{-1}\cdot\text{K}^{-1}$	měrná tepelná kapacita za konstantního objemu
$M$	$\text{kg}\cdot\text{kmol}^{-1}$	molární hmotnost
$m$	kg	hmotnost
$m_o$	kg	hmotnost molekuly
$M_r$	-	relativní molekulová hmotnost
$n$	mol	látkové množství
$n$	-	polytropický exponent
$p$	Pa	tlak
$Q$	J	teplo
$q$	$\text{J}\cdot\text{kg}^{-1}$	měrné teplo
$r$	$\text{J}\cdot\text{kg}^{-1}\cdot\text{K}^{-1}$	měrná plynová konstanta
$S$	J	entropie
$s$	$\text{J}\cdot\text{kg}^{-1}$	měrná entropie
$T$	K	termodynamická teplota
$V$	$\text{m}^3$	objem
$v$	$\text{m}^3\cdot\text{kg}^{-1}$	měrný objem
$K$	-	Poissonova konstanta

## Fyzikální konstanty

Symbol	Název konstanty
$k = 1,3806488 \cdot 10^{-23} \text{ J}\cdot\text{K}^{-1}$	Boltzmannova konstanta
$m_u = 1,66054 \cdot 10^{-27} \text{ kg}$	atomová hmotnostní konstanta
$N_A = 6,02214 \cdot 10^{23} \text{ mol}^{-1}$	Avogadrova konstanta
$R_m = 8314,472 \text{ J}\cdot\text{kmol}^{-1}\cdot\text{K}^{-1}$	univerzální plynová konstanta

## Indexy

Index	Význam
1	počáteční stav
2	konečný stav
12	děj mezi stavy 1 a 2

## **SEZNAM PŘÍLOH**

Příloha 1      Zdrojový text funkcí knihovny

## PŘÍLOHA 1 – ZDROJOVÝ TEXT FUNKCÍ KNIHOVNY

Tato příloha k bakalářské práci obsahuje výpis zdrojových textů všech m-souborů knihovny termodynamických funkcí, které byly vytvořeny v rámci práce. Jsou zde uspořádány v takovém pořadí, v jakém jsme se s nimi zabývali v práci.

### Volba pracovní látky – funkce *volplyn*:

```
% Funkce sloužící k volbě plynu
% Návod k volání funkce volplyn:
%   - po výzvě na displeji zadejte chemický vzorec molekuly
%   požadovaného plynu nebo "vzduch"
%   - výstupem funkce je sloupcový vektor plyn [molární hmotnost;
%   Poissonova konstanta]

function [plyn]=volplyn

% Tabulka plynů:
% chemický vzorec molekuly = [molární hmotnost; Poissonova konstanta]

Ar=[39.948;1.667];
He=[4.003;1.667];
H2=[2.016;1.4];
Ne=[20.183;1.667];
N2=[28.013;1.4];
O2=[31.999;1.393];
vzduch=[28.97;1.4];
NH3=[17.031;1.297];
CO2=[44.01;1.289];
CO=[28.01;1.399];
NO=[30.006;1.387];
N2O=[44.013;1.274];
NO2=[46.005;1.30];
R12=[120.914;1.126];
R22=[86.469;1.171];
R32=[52.024;1.242];
R125=[120.022;1.097];
R134a=[102.03;1.106];
SO2=[64.059;1.263];
SO3=[80.053;1.196];

plyn=input('Zvolte plyn (chem. znacka molekuly): ');

end
```

### Vlastnosti plynu – funkce *vlastn*:

```
% Funkce k určení r, cp, cv
% Návod k volání funkce vlastn:
%   - vstupním parametrem funkce je sloupcový vektor plyn
%   [molární hmotnost; Poissonova konstanta]
```

```
function [r,cp,cv]=vlastn(plyn)

Rm=8314.472;
r=Rm/(plyn(1,1));
cp=r/(1-1/plyn(2,1));
cv=cp-r;

end
```

## Stavová rovnice – funkce *str*:

```
% Funkce stavová rovnice - vrací hodnoty stavových veličin a
% hmotnosti m vypočítané ze stavové rovnice
% Vytvořeno: 11.2.2012
% Konzultace: 13.3.2012
% Návod k volání funkce str:
% - číselně zadány musí být min. 3 hodnoty z p, V, T, m
% - dále musí být zadána hodnota plyn
% - místo neznámé hodnoty zadejte mezi apostrofy libovolný textový
% řetězec (např. název hodnoty, text 'urci', apod.)
% - velmi důležité je dodržet pořadí zadávaných veličin, jejich
jednotky a rozměr (hodnota, vektor, matice):
% 1. p[Pa] - hodnota ... POZOR! TLAK V Pa NE v kPa (pozn. pouze
u této funkce a u funkce apstav)
% 2. V[m3], eventuelně v[m3/kg] - hodnota
% 3. T[K] - hodnota
% 4. plyn - sloupcový vektor: [molární hmotnost M[kg/kmol];
kappa[-]]
% (5. m[kg] - hodnota)
```

```
function [p,V,T,m]=str(varargin)

if nargin<4 % zadáno málo proměnných
disp('Pro výpočet nebyl zadán dostatečný počet vstupních
parametrů.')
```

```
disp('Musí být zadány minimálně čtyři vstupní parametry. Další
info viz. nápověda k funkci.')
```

```
elseif nargin>5 % zadáno moc proměnných
disp('Pro výpočet byl zadán nadbytečný počet vstupních
parametrů.')
```

```
disp('Může být zadáno maximálně pět vstupních parametrů. Další
info viz. nápověda k funkci.')
```

```
else
vstup=varargin;
p=vstup{1};
V=vstup{2};
T=vstup{3};
plyn=vstup{4};
if nargin==5
m=vstup{5}; % v tomto místě může být m číslo nebo řetězec,
když je řetězec, pak m určíme pomocí funkce
else
m=1;
end
```

```
% Základní podmínky řešitelnosti funkce (resp. stavové rovnice),
% ošetření výstupů

d=0;
if m==0
    d=1;
    disp('Byla zadána nulová hmotnost.')
    disp('Nelze určit žádné výstupní parametry.')
elseif T==0
    d=1;
    disp('Byla zadána nulová teplota.')
    disp('Nelze určit žádné výstupní parametry.')
elseif p==0
    d=1;
    disp('Byl zadán nulový tlak.')
    disp('Nelze určit žádné výstupní parametry.')
elseif V==0
    d=1;
    disp('Byl zadán nulový objem.')
    disp('Nelze určit žádné výstupní parametry.')
else

% VÝPOČET:
[r,cp,cv]=vlastn(plyn);

mat=[ischar(p) ischar(V) ischar(T) ischar(m)];
s=mat(1,1)+mat(1,2)+mat(1,3)+mat(1,4);

if s<=1

    nez=0;
    for i=1:4
        if mat(i)==1
            nez=i;
            break;
        end
    end

    switch nez
        case 1
            p=m*r*T/V;
        case 2
            V=m*r*T/p;
        case 3
            T=p*V/(m*r);
        case 4
            m=p*V/(r*T);
    end
else
    disp('Příliš mnoho neznámých ve stavové rovnici.')
end
end
end
end
```

**Převod tlaku z Pa na kPa – funkce *nakPa*:**

```
% Funkce pro převod tlaku z Pa na kPa
```

```
function [p]=nakPa(p)

if not(ischar(p))
    p=p/1000;
end

end
```

**Převod tlaku z kPa na Pa – funkce *naPa*:**

```
% Funkce pro převod tlaku z kPa na Pa
```

```
function [p]=naPa(p)

if not(ischar(p))
    p=p*1000;
end

end
```

**Převod teploty ze °C na K – funkce *naKel*:**

```
% Funkce pro převod teploty ze °C na K
```

```
function [T]=naKel(t)

if not(ischar(t))
    T=t+273.15;
end

end
```

**Převod teploty z K na °C – funkce *naCel*:**

```
% Funkce pro převod teploty z K na °C
```

```
function [t]=naCel(T)

if not(ischar(T))
    t=T-273.15;
end

end
```

**Izobarický děj – funkce *ib*:**

```
% Funkce k určení vektorů p, V, T, S (resp. p, v, T, s) izobarického
% děje a hmotnosti plynu m
% Návod k volání funkce ib:
% - číselně zadány musí být min. 3 hodnoty stavových veličin(p,V,T),
% přičemž nesmí nastat kombinace neznámých V1 a T1, resp. V2 a T2
% - dále musí být zadána hodnota plyn
% - na místo neznámé hodnoty zadejte mezi apostrofy libovolný
% textový řetězec (např. název hodnoty, text 'urci', apod.)
% - pokud je hmotnost plynu neznámou, kterou určujeme, je třeba,
% abychom aspoň v jednom stavu znali všechny stavové veličiny (nesmí
% nastat kombinace neznámých p a m)
% - na konci příkazu volání funkce je vhodné psát středník ";" a
% předejít tím zahlcení obrazovky (jinak budou zobrazeny 4 vektory
% p, V, T a S, každý po 300 hodnotách, spolu s hodnotou hmotnosti m)
% - velmi důležité je dodržet pořadí zadávaných veličin, jejich
% jednotky a rozměr (hodnota, vektor, matice):
%     1. p[kPa] - hodnota
%     2. V1[m3], ev. v1[m3/kg] - hodnota
%     3. T1[K] - hodnota
%     4. V2[m3], ev. v2[m3/kg] - hodnota
%     5. T2[K] - hodnota
%     6. plyn - sloupcový vektor: [molární hmotnost M[kg/kmol];
%     kappa[-]]
%     (7. m[kg] - hodnota)
```

```
function [p,V,T,S,m]=ib(varargin)
```

```
if nargin<6           % zadáno málo proměnných
    disp('Pro výpočet nebyl zadán dostatečný počet vstupních
parametrů.')
```

```
    disp('Musí být zadáno minimálně šest vstupních parametrů. Další
info viz nápověda k funkci.')
```

```
elseif nargin>7      % zadáno moc proměnných
    disp('Pro výpočet byl zadán přebytečný počet vstupních
parametrů.')
```

```
    disp('Může být zadáno maximálně sedm vstupních parametrů. Další
info viz nápověda k funkci.')
```

```
else
```

```
    vstup=varargin;
    p=vstup{1};
    V1=vstup{2};
    T1=vstup{3};
    V2=vstup{4};
    T2=vstup{5};
    plyn=vstup{6};
    if nargin==7
        m=vstup{7}; % v tomto místě může být m číslo nebo řetězec
    else
        m=1;
    end
```

```
[p]=naPa(p); % převod tlaku na Pa
```

```
[r,cp,cv]=vlastn(plyn);
```

```

% Matice zjištění neznámých proměnných
mat=[ischar(p) ischar(m); ischar(V1) ischar(T1); ischar(V2)
ischar(T2)];

s1=mat(1,1)+mat(1,2);
s2=mat(2,1)+mat(2,2);
s3=mat(3,1)+mat(3,2);

% Základní podmínky řešitelnosti funkce (resp. rovnic izobarického
% děje), ošetření výstupů

d=0;
if (s1+s2+s3)>2
    d=1;
    disp('Nebyl zadán dostatečný počet hodnot proměnných.')
    disp('Nelze určit žádné výstupní parametry.')
elseif (s1==2)|(s2==2)|(s3==2)|(s1==1&(s2+s3>1))
    d=1;
    disp('Nastala některá z nepovolených kombinací neznámých.')
    disp('Nelze určit žádné výstupní parametry.')
elseif not(ischar(m))&m==0
    d=1;
    disp('Byla zadána nulová hmotnost.')
    disp('Nelze určit žádné výstupní parametry.')
elseif (not(ischar(T1))&(T1==0))|(not(ischar(T1))&(T2==0))
    d=1;
    disp('Byla zadána nulová teplota.')
    disp('Nelze určit žádné výstupní parametry.')
elseif not(ischar(p))&p==0
    d=1;
    disp('Byl zadán nulový tlak.')
    disp('Nelze určit žádné výstupní parametry.')
elseif (not(ischar(V1))&(V1==0))|(not(ischar(V1))&(V2==0))
    d=1;
    disp('Byl zadán nulový objem.')
    disp('Nelze určit žádné výstupní parametry.')
else

% VÝPOČET:

if s1==1
    if s2==0
        [p1,V1,T1,m]=str(p,V1,T1,plyn,m);
        [p2,V2,T2,m]=str(p1,V2,T2,plyn,m);
    elseif s3==0
        [p2,V2,T2,m]=str(p,V2,T2,plyn,m);
        [p1,V1,T1,m]=str(p2,V1,T1,plyn,m);
    end
else
    [p1,V1,T1,m]=str(p,V1,T1,plyn,m);
    [p2,V2,T2,m]=str(p,V2,T2,plyn,m);
end

% Izobarický děj, musí platit p1=p2

p=linspace(p1,p2,300);

```

```

% V (resp. v) a T jsou na sobě závislé -> V naplním pomocí linspace,
% pro T musím sestavit funkci

V=linspace(V1,V2,300);
T(1)=T1;

for i=2:300
    T(i)=T(1)/V(1)*V(i);
end

for i=1:300
    S(i)=m*cp*log(T(i))-m*r*log(p(i));
end

[p]=nakPa(p);    % převod tlaku na kPa

end

if d
    p='Žádný výsledek';
    V='Žádný výsledek';
    T='Žádný výsledek';
    S='Žádný výsledek';
    m='Žádný výsledek';
end

end

```

## Izochorický děj – funkce *ich*:

```

% Funkce k určení vektorů p, V, T, S (resp. p, v, T, s) izochorického
% děje a hmotnosti plynu m
% Návod k volání funkce ich:
% - číselně zadány musí být min. 3 hodnoty stavových veličin(p,V,T),
%   přičemž nesmí nastat kombinace neznámých p1 a T1, resp. p2 a T2
% - dále musí být zadaná hodnota plyn
% - na místo neznámé hodnoty zadejte mezi apostrofy libovolný
%   textový řetězec (např. název hodnoty, text 'urci', apod.)
% - pokud je hmotnost plynu neznámou, kterou určujeme, je třeba,
%   abychom aspoň v jednom stavu znali všechny stavové veličiny (nesmí
%   nastat kombinace neznámých V a m)
% - na konci příkazu volání funkce je vhodné psát středník ";" a
%   předejít tím zahlcení obrazovky (jinak budou zobrazeny 4 vektory
%   p, V, T a S, každý po 300 hodnotách, spolu s hodnotou hmotnosti m)
% - velmi důležité je dodržet pořadí zadávaných veličin, jejich
%   jednotky a rozměr (hodnota, vektor, matice):
%   1. V[m3], ev. v[m3/kg] - hodnota
%   2. p1[kPa] - hodnota
%   3. T1[K] - hodnota
%   4. p2[kPa] - hodnota
%   5. T2[K] - hodnota
%   6. plyn - sloupcový vektor: [molární hmotnost M[kg/kmol] ;
%   kappa[-]]
%   (7. m[kg] - hodnota)

```

```

function [p,V,T,S,m]=ich(varargin)

if nargin<6          % zadáno málo proměnných
    disp('Pro výpočet nebyl zadán dostatečný počet vstupních
parametrů.')
```

disp('Musí být zadáno minimálně šest vstupních parametrů. Další
info viz nápověda k funkci.')

```
elseif nargin>7     % zadáno moc proměnných
    disp('Pro výpočet byl zadán přebytečný počet vstupních
parametrů.')
```

disp('Může být zadáno maximálně sedm vstupních parametrů. Další
info viz nápověda k funkci.')

```
else
    vstup=varargin;
    V=vstup{1};
    p1=vstup{2};
    T1=vstup{3};
    p2=vstup{4};
    T2=vstup{5};
    plyn=vstup{6};
    if nargin==7
        m=vstup{7}; % v tomto místě může být m číslo nebo řetězec
    else
        m=1;
    end

    [p1]=naPa(p1); % převod tlaku na Pa
    [p2]=naPa(p2);

    [r,cp,cv]=vlastn(plyn);

% Matice zjištění neznámých proměnných
    mat=[ischar(V) ischar(m); ischar(p1) ischar(T1); ischar(p2)
ischar(T2)];

    s1=mat(1,1)+mat(1,2);
    s2=mat(2,1)+mat(2,2);
    s3=mat(3,1)+mat(3,2);

% Základní podmínky řešitelnosti funkce (resp. rovnic izochorického
% děje), ošetření výstupů

    d=0;
    if (s1+s2+s3)>2
        d=1;
        disp('Nebyl zadán dostatečný počet hodnot proměnných.')
```

disp('Nelze určit žádné výstupní parametry.')

```
elseif (s1==2)|(s2==2)|(s3==2)|(s1==1&(s2+s3>1))
    d=1;
    disp('Nastala některá z nepovolených kombinací neznámých.')
```

disp('Nelze určit žádné výstupní parametry.')

```
elseif not(ischar(m))&m==0
    d=1;
    disp('Byla zadána nulová hmotnost.')
```

disp('Nelze určit žádné výstupní parametry.')

```
elseif (not(ischar(T1))&(T1==0)|(not(ischar(T1))&(T2==0))
    d=1;
    disp('Byla zadána nulová teplota.')
```

```

    disp('Nelze určit žádné výstupní parametry.')
elseif (not(ischar(p1))&(p1==0))|(not(ischar(p2))&(p2==0))
    d=1;
    disp('Byl zadán nulový tlak.')
    disp('Nelze určit žádné výstupní parametry.')
elseif not(ischar(V))&V==0
    d=1;
    disp('Byl zadán nulový objem.')
    disp('Nelze určit žádné výstupní parametry.')
else

% VÝPOČET:

    if s1==1
        if s2==0
            [p1,V1,T1,m]=str(p1,V,T1,plyn,m);
            [p2,V2,T2,m]=str(p2,V1,T2,plyn,m);
        elseif s3==0
            [p2,V2,T2,m]=str(p2,V,T2,plyn,m);
            [p1,V1,T1,m]=str(p1,V2,T1,plyn,m);
        end
    else
        [p1,V1,T1,m]=str(p1,V,T1,plyn,m);
        [p2,V2,T2,m]=str(p2,V,T2,plyn,m);
    end

% Izochorický děj, musí platit V1=V2

    V=linspace(V1,V2,300);

% p a T jsou na sobě závislé -> p naplním pomocí linspace, pro T musím
% sestavit funkci

    p=linspace(p1,p2,300);
    T(1)=T1;

    for i=2:300
        T(i)=T(1)/p(1)*p(i);
    end

    for i=1:300
        S(i)=m*cp*log(T(i))-m*r*log(p(i));
    end

    [p]=nakPa(p);    % převod tlaku na kPa

end

if d
    p='Žádný výsledek';
    V='Žádný výsledek';
    T='Žádný výsledek';
    S='Žádný výsledek';
    m='Žádný výsledek';
end

end

```

**Izotermický děj – funkce it:**

```

% Funkce k určení vektorů p, V, T, S (resp. p, v, T, s) izotermického
% děje a hmotnosti plynu m
% Návod k volání funkce it:
% - číselně zadány musí být min. 3 hodnoty stavových veličin(p,V,T),
% přičemž nesmí nastat kombinace neznámých p1 a V1, resp. p2 a V2
% - dále musí být zadaná hodnota plyn
% - na místo neznámé hodnoty zadejte mezi apostrofy libovolný
% textový řetězec (např. název hodnoty, text 'urci', apod.)
% - pokud je hmotnost plynu neznámou, kterou určujeme, je třeba,
% abychom aspoň v jednom stavu znali všechny stavové veličiny (nesmí
% nastat kombinace neznámých T a m)
% - na konci příkazu volání funkce je vhodné psát středník ";" a
% předejít tím zahlcení obrazovky (jinak budou zobrazeny 4 vektory
% p, V, T a S, každý po 300 hodnotách, spolu s hodnotou hmotnosti m)
% - velmi důležité je dodržet pořadí zadávaných veličin, jejich
% jednotky a rozměr (hodnota, vektor, matice):
% 1. T[K] - hodnota
% 2. p1[kPa] - hodnota
% 3. V1[m3], ev. v1[m3/kg] - hodnota
% 4. p2[kPa] - hodnota
% 5. V2[m3], ev. v2[m3/kg] - hodnota
% 6. plyn - sloupcový vektor: [molární hmotnost M[kg/kmol];
% kappa[-]]
% (7. m[kg] - hodnota)

```

```

function [p,V,T,S,m]=it(varargin)

if nargin<6          % zadáno málo proměnných
    disp('Pro výpočet nebyl zadán dostatečný počet vstupních
parametrů.')
    disp('Musí být zadáno minimálně šest vstupních parametrů. Další
info viz nápověda k funkci.')

elseif nargin>7     % zadáno moc proměnných
    disp('Pro výpočet byl zadán přebytečný počet vstupních
parametrů.')
    disp('Může být zadáno maximálně sedm vstupních parametrů. Další
info viz nápověda k funkci.')

else
    vstup=varargin;
    T=vstup{1};
    p1=vstup{2};
    V1=vstup{3};
    p2=vstup{4};
    V2=vstup{5};
    plyn=vstup{6};
    if nargin==7
        m=vstup{7}; % v tomto místě může být m číslo nebo řetězec
    else
        m=1;
    end

    [p1]=naPa(p1); % převod tlaku na Pa
    [p2]=naPa(p2);

```

```

[r,cp,cv]=vlastn(plyn);

% Matice zjištění neznámých proměnných
mat=[ischar(T) ischar(m); ischar(p1) ischar(V1); ischar(p2)
ischar(V2)];

s1=mat(1,1)+mat(1,2);
s2=mat(2,1)+mat(2,2);
s3=mat(3,1)+mat(3,2);

% Základní podmínky řešitelnosti funkce (resp. rovnic izotermického
% děje), ošetření výstupů

d=0;
if (s1+s2+s3)>2
    d=1;
    disp('Nebyl zadán dostatečný počet hodnot proměnných.')
    disp('Nelze určit žádné výstupní parametry.')
elseif (s1==2)|(s2==2)|(s3==2)|(s1==1&(s2+s3>1))
    d=1;
    disp('Nastala některá z nepovolených kombinací neznámých.')
    disp('Nelze určit žádné výstupní parametry.')
elseif not(ischar(m))&m==0
    d=1;
    disp('Byla zadána nulová hmotnost.')
    disp('Nelze určit žádné výstupní parametry.')
elseif not(ischar(T))&T==0
    d=1;
    disp('Byla zadána nulová teplota.')
    disp('Nelze určit žádné výstupní parametry.')
elseif (not(ischar(p1))&(p1==0))|(not(ischar(p1))&(p2==0))
    d=1;
    disp('Byl zadán nulový tlak.')
    disp('Nelze určit žádné výstupní parametry.')
elseif (not(ischar(V1))&(V1==0))|(not(ischar(V1))&(V2==0))
    d=1;
    disp('Byl zadán nulový objem.')
    disp('Nelze určit žádné výstupní parametry.')
else

% VÝPOČET:

    if s1==1
        if s2==0
            [p1,V1,T1,m]=str(p1,V1,T,plyn,m);
            [p2,V2,T2,m]=str(p2,V2,T1,plyn,m);
        elseif s3==0
            [p2,V2,T2,m]=str(p2,V2,T,plyn,m);
            [p1,V1,T1,m]=str(p1,V1,T2,plyn,m);
        end
    else
        [p1,V1,T1,m]=str(p1,V1,T,plyn,m);
        [p2,V2,T2,m]=str(p2,V2,T,plyn,m);
    end

% Izotermický děj, musí platit T1=T2

T=linspace(T1,T2,300);

```

```
% p a V jsou na sobě závislé -> p naplním pomocí linspace, pro V musím
% sestavit funkci
```

```
p=linspace(p1,p2,300);
V(1)=V1;

for i=2:300
    V(i)=p(1)*V(1)/p(i);
end

for i=1:300
    S(i)=m*cp*log(T(i))-m*r*log(p(i));
end

[p]=nakPa(p);    % převod tlaku na kPa

end

if d
    p='Žádný výsledek';
    V='Žádný výsledek';
    T='Žádný výsledek';
    S='Žádný výsledek';
    m='Žádný výsledek';
end

end
```

## Adiabatický děj – funkce *ad*:

```
% Funkce k určení vektorů p, V, T, S (resp. p, v, T, s) adiabatického
% děje a hmotnosti plynu m
% Návod k volání funkce ad:
% - počet neznámých nesmí být větší než 3
% - je-li m známá a počet neznámých je 3, nesmí být tyto neznámé v
% jednom stavu
% - je-li m neznámá, nesmí nastat žádná z kombinací neznámých p1 a
% p2, V1 a V2 nebo T1 a T2
% - dále musí být zadána hodnota plyn
% - na místo neznámé hodnoty zadejte mezi apostrofy libovolný
% textový řetězec (např. název hodnoty, text 'urci', apod.)
% - na konci příkazu volání funkce je vhodné psát středník ";" a
% předejít tím zahlcení obrazovky (jinak budou zobrazeny 4 vektory
% p, V, T a S, každý po 300 hodnotách, spolu s hodnotou hmotnosti m)
% - velmi důležité je dodržet pořadí zadávaných veličin, jejich
% jednotky a rozměr (hodnota, vektor, matice):
% 1. p1[kPa] - hodnota
% 2. V1[m3], ev. v1[m3/kg] - hodnota
% 3. T1[K] - hodnota
% 4. p1[kPa] - hodnota
% 5. V2[m3], ev. v2[m3/kg] - hodnota
% 6. T2[K] - hodnota
% 7. plyn - sloupcový vektor: [molární hmotnost M[kg/kmol];
% kappa[-]]
% (8. m[kg] - hodnota)
```

```

function [p,V,T,S,m]=ad(varargin)

if nargin<7          % zadáno málo proměnných
disp('Pro výpočet nebyl zadán dostatečný počet vstupních parametrů.')
disp('Musí být zadáno minimálně sedm vstupních parametrů. Další info
viz nápověda k funkci.')
```

d=1;

```

elseif nargin>8     % zadáno moc proměnných
disp('Pro výpočet byl zadán přebytečný počet vstupních parametrů.')
disp('Může být zadáno maximálně osm vstupních parametrů. Další info
viz nápověda k funkci.')
```

else

```

    vstup=varargin;
    p1=vstup{1};
    V1=vstup{2};
    T1=vstup{3};
    p2=vstup{4};
    V2=vstup{5};
    T2=vstup{6};
    plyn=vstup{7};
    if nargin==8
        m=vstup{8}; % v tomto místě může být m číslo nebo řetězec
    else
        m=1;
    end

    [p1]=naPa(p1); % převod tlaku na Pa
    [p2]=naPa(p2);

    [r,cp,cv]=vlastn(plyn);

% Matice identifikace neznámých stavových veličin (1. řádek - stav 1,
% 2. řádek - stav 2)
    mat=[ischar(p1) ischar(V1) ischar(T1); ischar(p2) ischar(V2)
ischar(T2)];

% určení počtu neznámých stavových veličin ve stavu 1
    s1=mat(1,1)+mat(1,2)+mat(1,3);
% určení počtu neznámých stavových veličin ve stavu 2
    s2=mat(2,1)+mat(2,2)+mat(2,3);

% Základní podmínky řešitelnosti funkce (resp. rovnic adiabatického
% děje), ošetření výstupů

    d=0;
    if (s1+s2)>3
        d=1;
        disp('Nebyl zadán dostatečný počet hodnot proměnných.')
        disp('Nelze určit žádné výstupní parametry.')
```

elseif (s1==3)|(s2==3)

```

    d=1;
    disp('Nebyly zadány hodnoty proměnných jednoho ze stavů.')
```

disp('Nelze určit žádné výstupní parametry.')

```

elseif not(ischar(m))&m==0
    d=1;
    disp('Byla zadána nulová hmotnost.')
```

disp('Nelze určit žádné výstupní parametry.')

```

elseif (not(ischar(T1))&(T1==0))|(not(ischar(T1))&(T2==0))
    d=1;
    disp('Byla zadána nulová teplota.')
    disp('Nelze určit žádné výstupní parametry.')
elseif (not(ischar(p1))&(p1==0))|(not(ischar(p1))&(p2==0))
    d=1;
    disp('Byl zadán nulový tlak.')
    disp('Nelze určit žádné výstupní parametry.')
elseif (not(ischar(V1))&(V1==0))|(not(ischar(V1))&(V2==0))
    d=1;
    disp('Byl zadán nulový objem.')
    disp('Nelze určit žádné výstupní parametry.')
else

    k=plyn(2,1);    % hodnota kappa

% VÝPOČET:

% Pro neznámou hmotnost m:
    if ischar(m)

% Ve stavu 1 známe všechny stavové veličiny (jedinou neznámou je m)
        if s1==0
            [p1,V1,T1,m]=str(p1,V1,T1,plyn,m); % určíme hmotnost m
            [p2,V2,T2]=apstav(p1,V1,T1,p2,V2,T2,plyn,k,m);

% Ve stavu 2 známe všechny stavové veličiny (jedinou neznámou je m)
% a ve stavu 1 existuje 1 nebo 2 neznámé stavové veličiny
            elseif (s2==0)&(s1~=0)
                [p2,V2,T2,m]=str(p2,V2,T2,plyn,m); % určíme hmotnost m
                [p1,V1,T1]=apstav(p2,V2,T2,p1,V1,T1,plyn,k,m);

% Ve stavu 1 i ve stavu 2 je po jedné neznámé stavové veličině
            elseif s1==s2

% Určení, která stavová veličina je ve stavu 1 neznámá
                for i=1:3
                    if mat(1,i)==1
                        break;
                    end
                end

% Určení, která stavová veličina je ve stavu 2 neznámá
                for j=1:3
                    if mat(2,j)==1
                        break;
                    end
                end

                switch i
                    case 1    % neznámý p1
                        switch j
                            case 1 % neznámý p2, nelze jej spočítat
                                disp('Neznáme hmotnost m, tlak p1 ani tlak p2.')
                                disp('Pomocí Vámi zadaných hodnot nelze určit žádné výstupní
parametry.')
                                    d=1;
                                case 2 % neznámý V2, známá T2
                                    p1=p2*(T1/T2)^(k/(k-1));
                                case 3 % neznámá T2, známý V2

```

```

        p1=p2*(V2/V1)^k;
    end
    case 2 % neznámý V1
        switch j
            case 1 % neznámý p2, známá T2
                V1=V2*(T2/T1)^(1/(k-1));
            case 2 % neznámý V2, nelze jej spočítat
                disp('Neznáme hmotnost m, objem V1 ani objem V2.')
                disp('Pomocí Vámi zadaných hodnot nelze určit žádné výstupní
parametry.')

```

```

end

    [p]=nakPa(p);    % převod tlaku na kPa
end

% Ošetření výstupů při nedodržení podmínek řešitelnosti
if d
    p='Žádný výsledek';
    V='Žádný výsledek';
    T='Žádný výsledek';
    S='Žádný výsledek';
    m='Žádný výsledek';
end

end

```

### Polytropický děj – funkce *pol*:

```

% Funkce k určení vektorů p, V, T, S (resp. p, v, T, s) polytropického
% děje a hmotnosti plynu m
% Návod k volání funkce pol:
% - počet neznámých nesmí být větší než 3
% - je-li n neznámou, musí být zadány hodnoty dvou stavových veličin
% v obou stavech, tedy p1, p2, T1 a T2 nebo p1, p2, V1 a V2 nebo
% V1, V2, T1 a T2
% - je-li m známá a počet neznámých je 3, nesmí být tyto neznámé v
% jednom stavu
% - je-li m neznámá, nesmí nastat žádná z kombinací neznámých
% p1 a p2, V1 a V2 nebo T1 a T2
% - dále musí být zadána hodnota plynu
% - na místo neznámé hodnoty zadejte mezi apostrofy libovolný
% textový řetězec (např. název hodnoty, text 'urci', apod.)
% - na konci příkazu volání funkce je vhodné psát středník ";" a
% předejít tím zahlcení obrazovky (jinak budou zobrazeny 4 vektory
% p, V, T a S, každý po 300 hodnotách, spolu s hodnotou hmotnosti m)
% - velmi důležité je dodržet pořadí zadávaných veličin, jejich
% jednotky a rozměr (hodnota, vektor, matice):
% 1. p1[kPa] - hodnota
% 2. V1[m3], ev. v1[m3/kg] - hodnota
% 3. T1[K] - hodnota
% 4. p2[kPa] - hodnota
% 5. V2[m3], ev. v2[m3/kg] - hodnota
% 6. T2[K] - hodnota
% 7. plyn - sloupcový vektor: [molární hmotnost M[kg/kmol]; %
% kappa[-]]
% 8. n[-] - hodnota
% (9. m[kg] - hodnota)

```

```
function [p,V,T,S,m]=pol(varargin)
```

```

if nargin<8          % zadáno málo proměnných
    disp('Pro výpočet nebyl zadán dostatečný počet vstupních
parametrů.')
    disp('Musí být zadáno minimálně osm vstupních parametrů. Další
info viz nápověda k funkci.')
    d=1;

```

```

elseif nargin>9      % zadáno moc proměnných
    disp('Pro výpočet byl zadán přebytečný počet vstupních
parametrů.')
```

```

    disp('Může být zadáno maximálně devět vstupních parametrů. Další
info viz nápověda k funkci.')
```

```

else
    vstup=varargin;
    p1=vstup{1};
    V1=vstup{2};
    T1=vstup{3};
    p2=vstup{4};
    V2=vstup{5};
    T2=vstup{6};
    plyn=vstup{7};
    n=vstup{8};
    if nargin==9
        m=vstup{9};    % v tomto místě může být m číslo nebo řetězec
    else
        m=1;
    end

    [p1]=naPa(p1);      % převod tlaku na Pa
    [p2]=naPa(p2);

    [r,cp,cv]=vlastn(plyn);

% Matice identifikace neznámých stavových veličin (1. řádek - stav 1,
% 2. řádek - stav 2)
    mat=[ischar(p1) ischar(V1) ischar(T1); ischar(p2) ischar(V2)
ischar(T2)];
% určení počtu neznámých stavových veličin ve stavu 1
    s1=mat(1,1)+mat(1,2)+mat(1,3);
% určení počtu neznámých stavových veličin ve stavu 2
    s2=mat(2,1)+mat(2,2)+mat(2,3);

    d=0;    % proměnná sloužící k ověření řešitelnosti funkce

% Výpočet polytropického exponentu n (v případě, že neznáme jeho
% hodnotu)
    if ischar(n)
        % zjištění, který vztah lze pro výpočet n použít
        vekn(1)=mat(1,1)+mat(1,2)+mat(2,1)+mat(2,2);
        vekn(2)=mat(1,2)+mat(1,3)+mat(2,2)+mat(2,3);
        vekn(3)=mat(1,1)+mat(1,3)+mat(2,1)+mat(2,3);
        for i=1:3
            if vekn(i)==0
                break;
            end
        end
        end
        if i<3
            % podle hodnoty i (určí známé proměnné) použijeme konkrétní
            % formu vztahu pro výpočet hodnoty n
            switch i
                case 1
                    n=log10(p1/p2)/log10(V2/V1);
                case 2
                    n=(log10(T1/T2)/log10(V2/V1))+1;
                case 3
                    n=log10(p2/p1)/(log10(p2/p1)-log10(T2/T1));
            end
        end
    end
end

```

```

        end
    else
        disp('Pomocí Vámi zadaných hodnot nelze vypočítat hodnotu
exponentu polytropy.')
        d=1;
    end
end

% Základní podmínky řešitelnosti funkce (resp. rovnic polytropického
% děje), ošetření výstupů

if (s1+s2)>3
    d=1;
    disp('Nebyl zadán dostatečný počet hodnot proměnných.')
    disp('Nelze určit žádné výstupní parametry.')
elseif (s1==3)|(s2==3)
    d=1;
    disp('Nebyly zadány hodnoty proměnných jednoho ze stavů.')
    disp('Nelze určit žádné výstupní parametry.')
elseif not(ischar(m))&m==0
    d=1;
    disp('Byla zadána nulová hmotnost.')
    disp('Nelze určit žádné výstupní parametry.')
elseif (not(ischar(T1))&(T1==0))|(not(ischar(T1))&(T2==0))
    d=1;
    disp('Byla zadána nulová teplota.')
    disp('Nelze určit žádné výstupní parametry.')
elseif (not(ischar(p1))&(p1==0))|(not(ischar(p1))&(p2==0))
    d=1;
    disp('Byl zadán nulový tlak.')
    disp('Nelze určit žádné výstupní parametry.')
elseif (not(ischar(V1))&(V1==0))|(not(ischar(V1))&(V2==0))
    d=1;
    disp('Byl zadán nulový objem.')
    disp('Nelze určit žádné výstupní parametry.')
elseif d~=1

% VÝPOČET:

% Pro neznámou hmotnost m
    if ischar(m)

% Ve stavu 1 známe všechny stavové veličiny (jedinou neznámou je m)
        if s1==0
            [p1,V1,T1,m]=str(p1,V1,T1,plyn,m); % určíme hmotnost m
            [p2,V2,T2]=apstav(p1,V1,T1,p2,V2,T2,plyn,n,m);

% Ve stavu 2 známe všechny stavové veličiny (jedinou neznámou je m)
% a ve stavu 1 existuje 1 nebo 2 neznámé stavové veličiny
            elseif (s2==0)&(s1~=0)
                [p2,V2,T2,m]=str(p2,V2,T2,plyn,m); % určíme hmotnost m
                [p1,V1,T1]=apstav(p2,V2,T2,p1,V1,T1,plyn,n,m);

% Ve stavu 1 i ve stavu 2 je po jedné neznámé stavové veličině
            elseif s1==s2

% Určení, která stavová veličina je ve stavu 1 neznámá
                for i=1:3
                    if mat(1,i)==1
                        break;

```

```

        end
    end

% Určení, která stavová veličina je ve stavu 2 neznámá
for j=1:3
    if mat(2,j)==1
        break;
    end
end

switch i
    case 1 % neznámý p1
        switch j
            case 1 % neznámý p2, nelze jej spočítat
                disp('Neznáme hmotnost m, tlak p1 ani tlak p2.')
                disp('Pomocí Vámi zadaných hodnot nelze určit žádné výstupní
parametry.')
            case 2 % neznámý V2, známá T2
                d=1;
                p1=p2*(T1/T2)^(n/(n-1));
            case 3 % neznámá T2, známý V2
                p1=p2*(V2/V1)^n;
        end
    case 2 % neznámý V1
        switch j
            case 1 % neznámý p2, známá T2
                V1=V2*(T2/T1)^(1/(n-1));
            case 2 % neznámý objem V2, nelze jej
spočítat
                disp('Neznáme hmotnost m, objem V1 ani objem V2.')
                disp('Pomocí Vámi zadaných hodnot nelze určit žádné výstupní
parametry.')
            case 3 % neznámá T2, známý p2
                d=1;
                V1=V2*(p2/p1)^(1/n);
        end
    case 3 % neznámá T1
        switch j
            case 1 % neznámý p2
                T1=T2*(V2/V1)^(n-1);
            case 2 % neznámý V2
                T1=T2*(p1/p2)^((n-1)/n);
            case 3 % neznámá teplota T2
                disp('Neznáme hmotnost m, teplotu T1 ani teplotu T2.')
                disp('Pomocí Vámi zadaných hodnot nelze určit žádné výstupní
parametry.')
        end
    case 4 % neznámá T1
        d=1;
    end
end

[p1,V1,T1,m]=str(p1,V1,T1,plyn,m);
[p2,V2,T2,m]=str(p2,V2,T2,plyn,m);
end

% Pro známou hmotnost m
else
    if s1<=s2
        % V tomto případě bude ve stavu 1 vždy max. 1 neznámá
        [p1,V1,T1,m]=str(p1,V1,T1,plyn,m);
        [p2,V2,T2]=apstav(p1,V1,T1,p2,V2,T2,plyn,n,m);
    else

```

```

        % V tomto případě bude ve stavu 2 vždy max. 1 neznámá
        [p2,V2,T2,m]=str(p2,V2,T2,plyn,m);
        [p1,V1,T1]=apstav(p2,V2,T2,p1,V1,T1,plyn,n,m);
    end
end

% Plnění výstupních vektorů
% tlak
p=linspace(p1,p2,300);
% teplota
T(1)=T1;
for i=2:300
    T(i)=T(1)*(p(i)/p1)^((n-1)/n);
end
% objem
V(1)=V1;
for i=2:300
    V(i)=V(1)*(T1/T(i))^(1/(n-1));
end
% entropie
for i=1:300
    S(i)=m*cp*log(T(i))-m*r*log(p(i));
end

[p]=nakPa(p); % převod tlaku na kPa
end

% Ošetření výstupů při nedodržení podmínek řešitelnosti
if d
    p='Žádný výsledek';
    V='Žádný výsledek';
    T='Žádný výsledek';
    S='Žádný výsledek';
    m='Žádný výsledek';
end

end

```

## Funkce *apstav*:

```

% Funkce apstav ze znalosti všech stavových veličin jednoho stavu
% polytropického (nebo adiabatického) děje, hmotnosti plynu,
% hodnoty n (kappa), a ze znalosti minimálně jedné hodnoty ve stavu
% druhém, určí zbývající stavové veličiny ve druhém stavu tohoto děje
% Použití: především funkce ad a pol
% Návod k volání funkce:
% - musí být zadány hodnoty p1, V1, T1, plyn, n, m a jedna z hodnot
% p2, V2 nebo T2
% - místo neznámé hodnoty zadejte mezi apostrofy libovolný textový
% řetězec (např. název hodnoty, text 'urci', apod.)
% - velmi důležité je dodržet pořadí zadávaných veličin, jejich
% jednotky a rozměr (hodnota, vektor, matice):
% 1. p1[Pa] - hodnota ... POZOR! TLAK V Pa NE v kPa
% (pozn. pouze u této funkce a u funkce apstav)
% 2. V1[m3], eventuelně v1[m3/kg] - hodnota
% 3. T1[K] - hodnota
% 4. p2[Pa] - hodnota ... POZOR! TLAK V Pa NE v kPa
% 5. V2[m3], eventuelně v2[m3/kg] - hodnota
% 6. T2[K] - hodnota

```

```

%       7. plyn - sloupcový vektor: [molární hmotnost M[kg/kmol]];
%       kappa[-]
%       8. n[-] - hodnota
%       9. m[kg] - hodnota

function [p2,V2,T2]=apstav(p1,V1,T1,p2,V2,T2,plyn,n,m)

% Vektor indikace neznámých stavu 2
vek=[ischar(p2) ischar(V2) ischar(T2)];

nez=0;
for i=1:length(vek)
    nez=nez+vek(i);    % TADY TOTO JSEM OPRAVOVAL, PŘEDTÍM TU BYLO
nez=nez+1;
end

switch nez
    case 1                % ve stavové rovnici pro stav 1 je 1 neznámá
        [p2,V2,T2,m]=str(p2,V2,T2,plyn,m);
    case 2                % ve stavové rovnici pro stav 2 jsou 2 neznámé
        for i=1:3        % určení, která proměnná je známá
            if vek(i)==0
                break;
            end
        end
        switch i
            case 1        % známe p2
                T2=T1*(p2/p1)^((n-1)/n);
            case 2        % známe V2
                T2=T1*(V1/V2)^(n-1);
            case 3        % známe T2
                p2=p1*(T2/T1)^(n/(n-1));
            end
        [p2,V2,T2,m]=str(p2,V2,T2,plyn,m);
    otherwise
end
end

```

### Funkce *stavy*:

```

% Funkce slouží k vypsání stavových veličin v krajních stavech děje
% určeného vektory p, V, T, S (resp. p, v, T, s) do proměnných
% Návod k volání funkce:
% - zadány musí být p, V (v), T a S (s) ve formě řádkových vektorů
% - tip: k vytvoření vektorů užíjte funkcí pro jednotlivé děje (ib,
% it, ad, apod.)

```

```

function [p1,p2,V1,V2,T1,T2,S1,S2]=stavy(p,V,T,S)

% Stav 1:

p1=p(1);
V1=V(1);
T1=T(1);
S1=S(1);

```

```
% Stav 2:

p2=p(length(p));
V2=V(length(V));
T2=T(length(T));
S2=S(length(S));

end
```

### **Funkce *stavyd*:**

```
% Funkce stavyd slouží k okamžitému vypsání stavových veličin v
% krajních stavech děje určeného vektory p, V, T, S (resp. p, v, T, s)
% a hmotnosti plynu na displej
% Návod k volání funkce:
% - zadány musí být p, V (v), T a S (s) ve formě řádkových vektorů
% - tip: k vytvoření vektorů užíjte funkci pro jednotlivé děje (ib,
% it, ad apod.)
```

```
function stavyd(p,V,T,S)
```

```
% Stav 1:
```

```
p1=p(1);
V1=V(1);
T1=T(1);
S1=S(1);
```

```
fprintf('STAV 1: \n - tlak[kPa]: %f \n - objem; měrný objem[m3;
m3/kg]: %f \n - teplota[K] %f \n - entropie[J]: %f \n\n',p1,V1,T1,S1')
```

```
% Stav 2:
```

```
p2=p(length(p));
V2=V(length(V));
T2=T(length(T));
S2=S(length(S));
```

```
fprintf('STAV 2: \n - tlak[kPa]: %f \n - objem; měrný objem[m3;
m3/kg]: %f \n - teplota[K] %f \n - entropie[J]: %f \n\n',p2,V2,T2,S2')
```

```
end
```

## Objemová práce – funkce *praceo*:

```
% Funkce k určení objemové práce (použitelná pro 1kg látky i pro mkg
% látky)
% Návod k volání funkce praceo:
% - zadány musí být p a V (resp. p a v) ve formě řádkových vektorů
% - tip: k vytvoření vektorů užíjte funkcí pro jednotlivé děje (ib,
% it, ad, apod.)
```

```
function [ao]=praceo(p,V)

[p]=naPa(p);
ao=0;
for k=1:299
    ao=ao+((p(k)+p(k+1))/2*(V(k+1)-V(k)));
end
end
```

## Technická práce – funkce *pracet*:

```
% Funkce k určení technické práce (použitelná pro 1kg látky i pro mkg
% látky)
% Návod k volání funkce pracet:
% - zadány musí být p a V (resp. p a v) ve formě řádkových vektorů
% - tip: k vytvoření vektorů užíjte funkcí pro jednotlivé děje (ib,
% it, ad, apod.)
```

```
function [at]=pracet(p,V)

[p]=naPa(p);
at=0;
for k=1:299
    at=at+((V(k)+V(k+1))/2*(p(300-k+1)-p(300-k)));
end
at=-at;          % znaménková konvekce
end
```

## Teplo – funkce *teplo*:

```
% Funkce k určení tepla (použitelná pro 1kg látky i pro mkg látky)
% Návod k volání funkce teplo:
% - zadány musí být T a S (resp. T a s) ve formě řádkových vektorů
% - tip: k vytvoření vektorů užíjte funkcí pro jednotlivé děje (ib,
% it, ad, apod.)
```

```
function [q12]=teplo(T,S)

q12=0;
for k=1:299
    q12=q12+((T(k)+T(k+1))/2*(S(k+1)-S(k)));
end
end
```

## Vykreslení p-V (p-v) diagramu – funkce *pvdiag*:

```
% Funkce pro vykreslení p-V (resp. p-v) diagramu termodynamického
% cyklu případně děje
% Návod k volání funkce pvdiag:
% - zadány musí být mat_p a mat_V (resp. mat_p a mat_v) ve formě
% matic, kde jednotlivé řádky představují děje cyklu, který chceme
% v p-V diagramu zobrazit, důležité je, aby obě matice měly stejný
% počet řádků
% - v případě, že chceme zobrazit pouze děj, mohou být vstupní
% parametry ve formě řádkových vektorů p a V (resp. p a v)
% - dále musí být zadána hodnota plyn - sloupcový vektor [molární
% hmotnost M[kg/kmol] ; kappa[-]]
% - tip: k vytvoření vektorů dějů užíjte funkce pro jednotlivé děje
% (ib, it, ad, apod.)
```

```
function pvdiag(mat_p,mat_V,plyn)

% Zjištění, zda byl zadán objem nebo měrný objem
merny=input('OBJEM V nebo MĚRNÝ OBJEM v? [zadejte textový řetězec "V"
nebo "v"]: ');
merny=strcmp(merny,'v');
S_p=size(mat_p);
S_V=size(mat_V);

if S_p(1)==S_V(1)

    for i=1:S_p(1)

        plot(mat_V(i,:),mat_p(i,:));
        hold on

        % Krajní stavy děje - označení
        plot(mat_V(i,1),mat_p(i,1),'bo')
        plot(mat_V(i,300),mat_p(i,300),'bo')

        % Popis
        if mat_V(i,1)<mat_V(i,2)
            D=1;
            % Popis krajních stavů v diagramu
            text(mat_V(i,1)-mat_V(i,1)/8,mat_p(i,1),num2str(i))
            text(mat_V(i,300)+mat_V(i,1)/8,mat_p(i,300),num2str(i+1))
        elseif mat_V(i,1)>mat_V(i,2)
            D=1;
            % Popis krajních stavů v diagramu
            text(mat_V(i,1)+mat_V(i,300)/8,mat_p(i,1),num2str(i))
            text(mat_V(i,300)-
mat_V(i,300)/8,mat_p(i,300),num2str(i+1))
        else
            dej='Izochorický děj';
            D=0;
            % Popis krajních stavů v diagramu
            text(mat_V(i,1)-mat_V(i,1)/8,mat_p(i,1),num2str(i))
            text(mat_V(i,1)-mat_V(i,1)/8,mat_p(i,300),num2str(i+1))
        end

    end

end
```

```

% Popis os grafu
if merny
    xlabel('Měrný objem v [m3/kg]')
    titulek=' p-v DIAGRAM';
else
    xlabel('Objem V [m3]')
    titulek=' p-V DIAGRAM';
end
ylabel('Tlak p [kPa]')

% Jde-li o p-V diagram děje, uvedeme nad graf, o jaký typ
% termodynamického děje jde:

if i==1
% Zjištění, o jaký děj jde (v případě, že nejde o izochorický děj):
    if D
        n=log10(mat_p(1)/mat_p(300))/log10(mat_V(300)/mat_V(1));
        switch n
            case 0
                dej='Izobarický děj';
            case 1
                dej='Izotermický děj';
            case plyn(2,1)
                dej='Adiabatický děj';
            otherwise
                dej='Polytropický děj';
        end
        end
        % Popisek grafu
        title(char(titulek,dej))

    else
        % Popisek grafu
        title(char(titulek))

    end

    % Nastavení rozsahu os
    axis([0 max(max(mat_V))*1.2 0 max(max(mat_p))*1.2]);

    hold off

else
    disp('Byla zadána špatná forma vstupních parametrů, diagram nelze
zobrazit.')
end

end

```

## Vykreslení T-S (T-s) diagramu – funkce *tsdiag*:

```

% Funkce pro vykreslení T-S (resp. T-s) diagramu termodynamického
% cyklu případně děje
% Návod k volání funkce tsdiag:
% - zadány musí být mat_T a mat_S (resp. mat_T a mat_s) ve formě
% matic, kde jednotlivé řádky představují děje cyklu, který chceme
% v T-S diagramu zobrazit, důležité je, aby obě matice měly stejný
% počet řádků

```

```
% - v případě, že chceme zobrazit jen děj, mohou být vstupní
% parametry ve formě řádkových vektorů T a S (resp. T a s)
% - dále musí být zadána hodnota plyn - sloupcový vektor [molární
% hmotnost M[kg/kmol] ; kappa[-]]
% - povinným vstupním parametrem je také hodnota hmotnosti
% - tip: k vytvoření vektorů dějů užíjte funkce pro jednotlivé děje
% (ib, it, ad, apod.)
```

```
function tsdiag(mat_T,mat_S)
```

```
% Zjištění, zda byla zadána entropie nebo měrná entropie
merny=input('ENTROPIE S nebo MĚRNÁ ENTROPIE s? [zadejte textový
řetězec "S" nebo "s"]: ');
merny=strcmp(merny,'s');

S_T=size(mat_T);
S_S=size(mat_S);

if S_T(1)==S_S(1)

    for i=1:S_T(1)

        plot(mat_S(i,:),mat_T(i,:));
        hold on

        % Krajní stavy děje - označení
        plot(mat_S(i,1),mat_T(i,1),'bo')
        plot(mat_S(i,300),mat_T(i,300),'bo')

        a=round(mat_S(i,1));
        b=round(mat_S(i,2));

        % Popis krajních stavů v diagramu
        if a<b
            text(mat_S(i,1)-mat_S(i,1)/8,mat_T(i,1),num2str(i))
            text(mat_S(i,300)+mat_S(i,1)/8,mat_T(i,300),num2str(i+1))
        elseif a>b
            text(mat_S(i,1)+mat_S(i,300)/8,mat_T(i,1),num2str(i))
            text(mat_S(i,300)-
mat_S(i,300)/8,mat_T(i,300),num2str(i+1))
        else
            text(mat_S(i,1)-mat_S(i,1)/8,mat_T(i,1),num2str(i))
            text(mat_S(i,1)-mat_S(i,1)/8,mat_T(i,300),num2str(i+1))
        end

    end

    % Popis os grafu
    if merny
        xlabel('Měrná entropie s [J/kg]')
        titulek=' T-s DIAGRAM';
    else
        xlabel('Entropie S [J]')
        titulek=' T-S DIAGRAM';
    end

    ylabel('Teplota T [K]')
```

```
% Popisek grafu
title(char(titulek))

% Nastavení rozsahu os
axis([0 max(max(mat_S))*1.2 0 max(max(mat_T))*1.2]);

hold off

else
    disp('Byla zadána špatná forma vstupních parametrů, diagram nelze
zobrazit.')
end

end
```