

2. Algoritmy s vetvením

- vetvenie umožňuje pokračovať v programe rôznymi cestami
- na základe splnenia, resp. nesplnenia podmienky sa vyberajú ďalšie kroky programu
- umožňuje ošetriť program aby vypísal chybovú hlášku pri nesprávnom zadaní vstupu
- poznáme niekoľko typov vetvenia:

preskok – pri splnení podmienky sa vykoná daný program, pri nesplnení sa program preskočí

dvojcestné/trojcestné vetvenie – vykoná sa jedna z dvoch/troch alternatív podľa podmienky

prepínač – má nekonečne veľa alternatív ako môže program pokračovať

1. Vytvorte algoritmus pre zistenie absolútnej hodnoty zadaného čísla.

Vstupná premenná: x

Výstupná premenná: AH

Analýza riešenia:

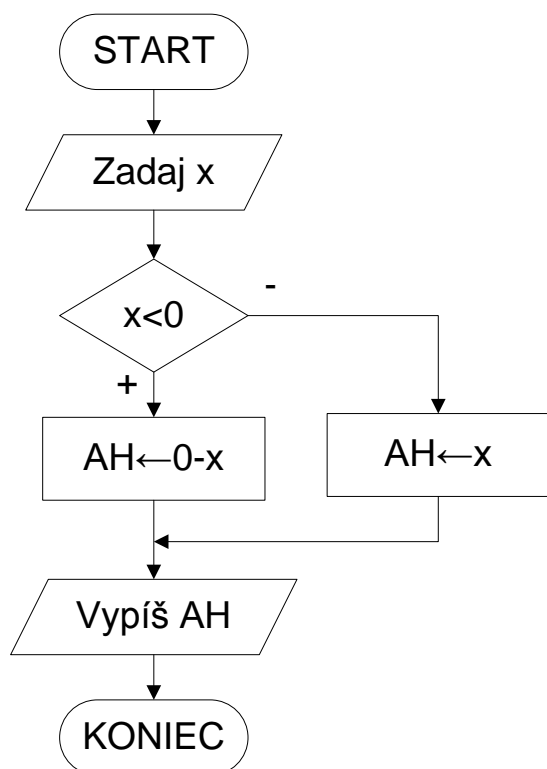
Absolútna hodnota čísla predstavuje vzdialenosť obrazu čísla na číselnej osi od čísla 0.

Absolútna hodnota je vždy kladné číslo. Ak je číslo zadané na vstupe kladné, potom abs.

hodnota zadaného čísla je jeho hodnota, teda $AH=x$. Pri zápornom čísle ide o číslo opačné, teda absolútnu hodnotu môžeme zistiť viacerými spôsobmi, napr. odčítaním čísla x od čísla 0 alebo vynásobením čísla x číslom -1.

Slovný popis algoritmu:

1. krok: zadaj číslo x
2. krok: zistiť či je $x < 0$
3. krok: ak číslo $x < 0$ – prejsť na 4. krok
4. krok: $AH \leftarrow 0 - x$
5. krok: ak číslo $x \geq 0$ – prejsť na 6. krok
6. krok: $AH \leftarrow x$
7. krok: vytlačiť AH



```
int main(void)
{
    float x;
    float AH;

    printf("Absolutna hodnota cisla \n\n");

    printf("Zadaj cislo x: ");
    scanf("%f",&x);

    if(x<0){
        AH=0-x;
    }
    else{
        AH=x;
    }

    printf("\nAbs. hodnota cisla %0.2f je %0.2f. \n\n",x,AH);
    system("PAUSE");
    return 0;
}
```

2. Vytvorte algoritmus pre výpočet objemu kvádra, prípadne valca pričom užívateľ sám určí či chce počítať objem kvádra alebo valca zadáním čísla do premennej útvar (ut). Pre zjednodušenie použijeme pre premennú „ut“ číselné hodnoty. Ak užívateľ zadá číslo menšie ako 0 spustí sa program na výpočet objemu kvádra, ak zadané číslo bude vyššie alebo rovné nule, spustí sa program na výpočet objemu valca.

Vstupné premenné: ut, a, b, c, r, v

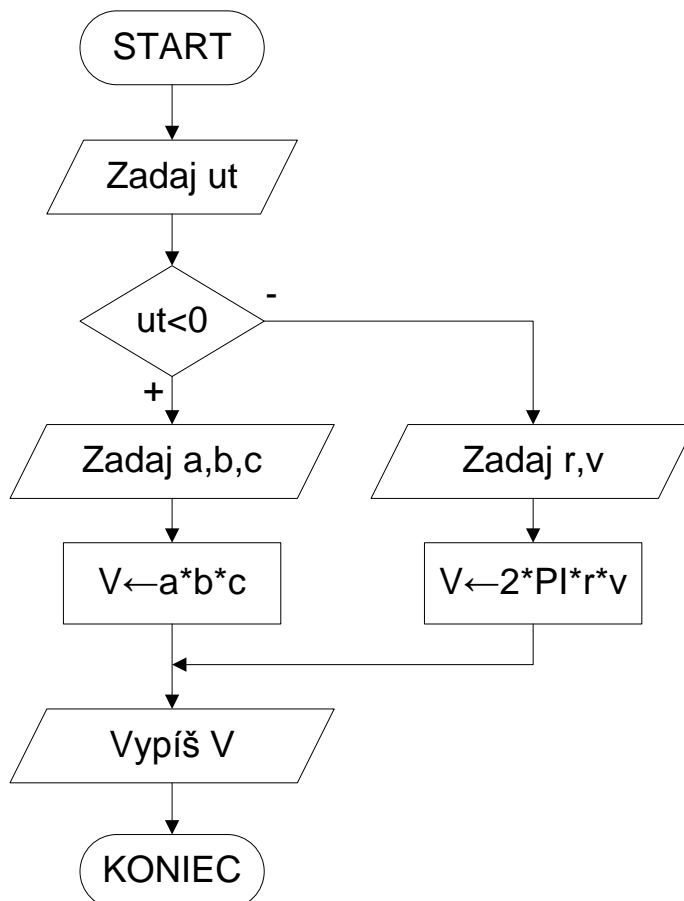
Výstupná premenná: V

Analýza riešenia:

Pred písaním samotnej funkcie si definujeme hodnotu premennej pí príkazom #define PI 3.1415926. Program vyzve užívateľa k zadaniu útvaru, ktorého objem chce počítať. Použijeme vetvenie aby si užívateľ mohol vybrať požadovaný program. Pre zjednodušenie sme použili pre „ut“ typ premennej int. Ak má „ut“ zápornú hodnotu, počíta sa objem kvádra, v inom prípade počítame objem valca.

Slovný popis algoritmu:

1. krok: zadaj číslo ut
2. krok: zistiť či je $ut < 0$
3. krok: ak je $ut < 0$ – prejsť na krok 4



```

#include <stdio.h>
#define PI 3.1415926

int main(void)
{
    int ut;
    float a,b,c,r,v,V;

    printf("Vypocet objemu kvadra(-)/valca(+) \n\n");

    printf("Zadaj utvar: ");
    scanf("%d",&ut);

    if(ut<0){
        printf("Zadaj sirku a v cm: ");
        scanf("%f",&a);
        printf("Zadaj dlzku b v cm: ");
        scanf("%f",&b);
        printf("Zadaj vysku c v cm: ");
        scanf("%f",&c);

        V=a*b*c;
        printf("\nObjem kvadra je %0.2f cm3. \n\n",V);
    }
    else{
        printf("Zadaj polomer r v cm: ");
        scanf("%f",&r);
        printf("Zadaj vysku v v cm: ");
        scanf("%f",&v);

        V=2*PI*r*v;

        printf("\nObjem valca je %0.2f cm3.\n\n",V);
    }

    system("PAUSE");
    return 0;
}

```

3. Vytvorte algoritmus, ktorý vypočíta neznámu stranu obdĺžnika. Užívateľ na vstupe zadá, ktorý parameter obdĺžnika pozná (obsah alebo obvod) a tiež, ktorá strana (a alebo b) je známa. Následne bude vyzvaný k zadaniu hodnoty známeho parametra, potom k zadaniu dĺžky známej strany. Pre uľahčenie použijeme na vstupe číselné hodnoty nasledovným spôsobom:

Parameter:

1 – obsah

2 – obvod

Strana:

1 – strana a

2 – strana b

Použitím trojcestného vetvenia ošetríte program aby pri zadaní čísla iného ako je 1 alebo 2 vypísal chybovú hlášku a následne sa ukončí.

Vstupné premenné: P, St, S/O, a/b

Výstupné premenné: a/b

Analýza riešenia:

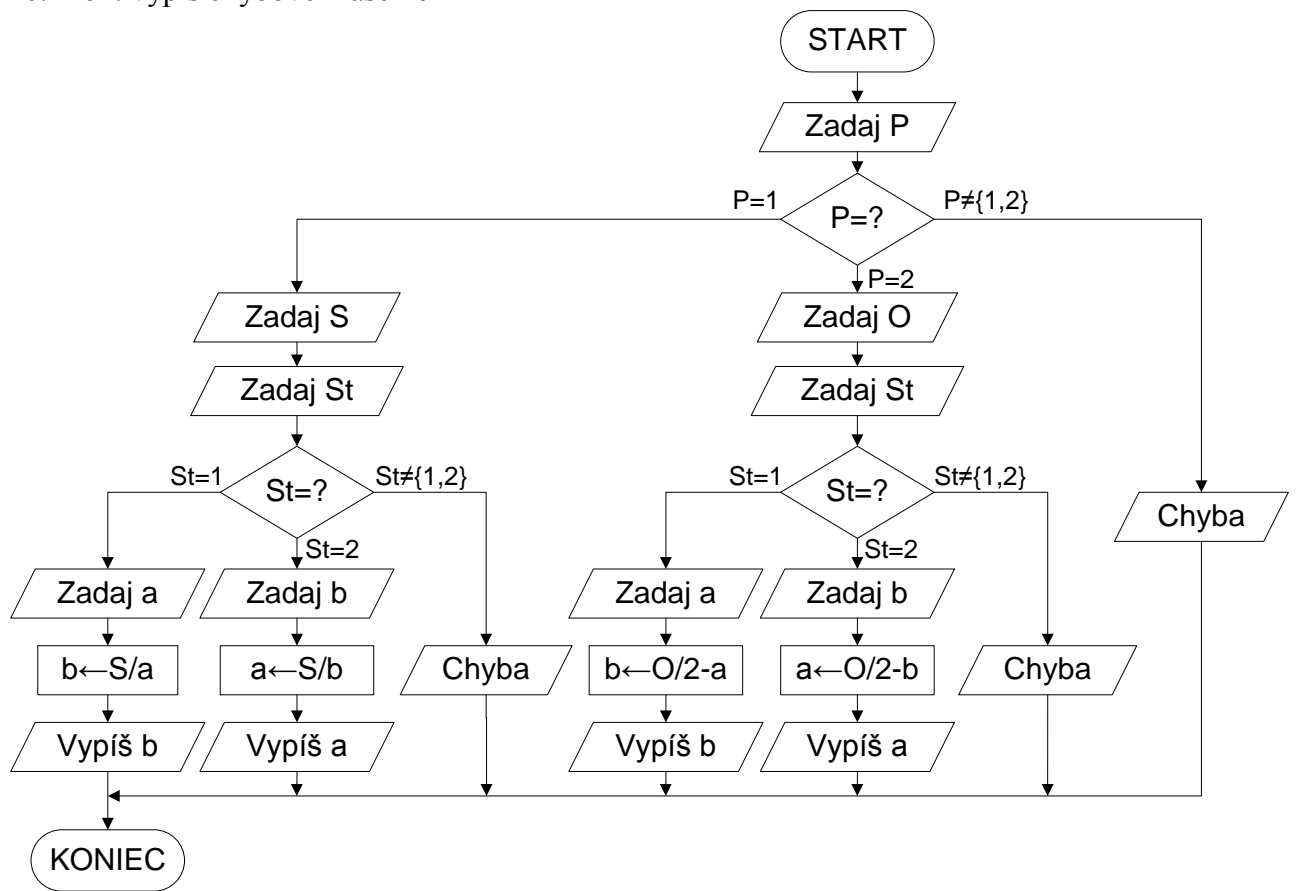
Program začne vyzvaním užívateľa k zadaniu číselnej hodnoty známeho parametra. Po zadaní nasleduje trojcestné vetvenie, ktoré vyzve užívateľa k zadaniu hodnoty daného parametra a následne k zadaniu čísla 1 alebo 2 podľa toho, ktorá strana je známa. Nasleduje ďalšie vetvenie, ktoré rozhodne o ďalšom postupe – buď vyzve užívateľa k zadaniu dĺžky strany a, ak je hodnota premennej $St=1$ alebo b, ak má premenná St hodnotu 2 alebo vypíše chybové hlásenie pri zadaní akéhokoľvek iného čísla. Po zadaní dĺžky strany zbehne určený program a program vypíše pôvodne neznámu stranu obdĺžnika.

Slovný popis algoritmu:

1. krok: zadaj známy parameter P
2. krok: ak $P=1$ pokračuj krokom 3, inak prejdi na krok 14
3. krok: zadaj obsah S
4. krok: zadaj známu stranu St
5. krok: ak $St=1$ pokračuj krokom 6, inak prejdi na krok 9
6. krok: zadaj dĺžku strany a
7. krok: výpočet strany $b=S/a$
8. krok: vypíš dĺžku strany b
9. krok: ak $St=2$ pokračuj krokom 10, inak prejdi na krok 13
10. krok: zadaj dĺžku strany b
11. krok: výpočet strany $a=S/b$
12. krok: vypíš dĺžku strany a
13. krok: vypíš chybové hlásenie
14. krok: ak $P=2$ pokračuj krokom 15, inak prejdi na krok 26
15. krok: zadaj obvod O
16. krok: zadaj známu stranu St
17. krok: ak $St=1$ pokračuj krokom 18, inak prejdi na krok 21
18. krok: zadaj dĺžku strany a
19. krok: výpočet strany $b=O/2-a$
20. krok: vypíš dĺžku strany b
21. krok: ak $St=2$ pokračuj krokom 22, inak prejdi na krok 25
22. krok: zadaj dĺžku strany b
23. krok: výpočet strany $a=O/2-b$
24. krok: vypíš dĺžku strany a

25. krok: vypíš chybové hlásenie

26. krok: vypíš chybové hlásenie



```

int main(void)
{
float S,O,a,b,P,St;

printf("Vypocet chybajucej strany obdlznika x \n\n");

printf("Zadaj znamy parameter (1-obsah/2-obvod): ");
scanf("%f",&P);

if(P==1){
printf("Zadaj obsah S: ");
scanf("%f",&S);
printf("Zadaj znamu stranu (1-strana a/2-strana b): ");
scanf("%f",&St);

if(St==1){
printf("Zadaj dlzku strany a v cm: ");
scanf("%f",&a);
b=S/a;
printf("\nDlзка strany b je %0.2f cm.\n\n",b);
}
else if(St==2){
printf("Zadaj dlzku strany b v cm: ");
scanf("%f",&b);
a=S/b;
printf("\nDlзка strany a je %0.2f cm.\n\n",a);
}
else printf("\nZadanie nebolo spravne!\n\n");
}

else if(P==2){
printf("Zadaj obvod O: ");
scanf("%f",&O);
printf("Zadaj znamu stranu (1-strana a/2-strana b): ");
scanf("%f",&St);

if(St==1){
printf("Zadaj dlzku strany a v cm: ");
scanf("%f",&a);
b=O/2-a;
printf("\nDlзка strany b je %0.2f cm.\n\n",b);
}
else if(St==2){
printf("Zadaj dlzku strany b v cm: ");
scanf("%f",&b);
a=O/2-b;
printf("\nDlзка strany a je %0.2f cm.\n\n",a);
}
else printf("\nZadanie nebolo spravne!\n\n");
}

else printf("\nZadanie nebolo spravne!\n\n");

system("PAUSE");
return 0;
}

```