

# ZAP-PM II.

Matej Gazda

# Motivácia

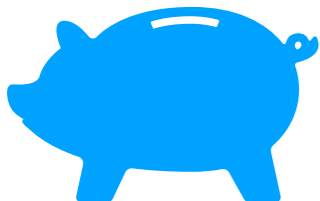
## Pochopiť:

- Syntax jazyka C, dátové typy, funkcie, if-else klauzula (while loop)
- Vývojové diagramy

## Naprogramovať:



Algoritmus na kupovanie / predávanie akcií (stocks)



# Motivácia

Pochopiť:

- Syntax jazyka C, dátové typy, funkcie, if-else klauzula (while loop)
- Vývojové diagramy

Napr



# Syntax jazyka C - datové typy

Najpoužívanéjšie:

Data Type	Memory (bytes)	Range	Format specifier
Short int	2	-32,768 to 32,767	%hd
Unsigned short int	2	0 to 65,535	%hu
Unsigned Char	1	0 to 255	%c
Int	4	-2,147,483,648 to 2,147,483,647	%d
Float	4		%f
Double	8		%lf

# Syntax jazyka C - dátové typy

```
#include <stdio.h>

int main() {
    int a = 2147483650;
    printf("%d", a);
}
```



Čo sa stane?

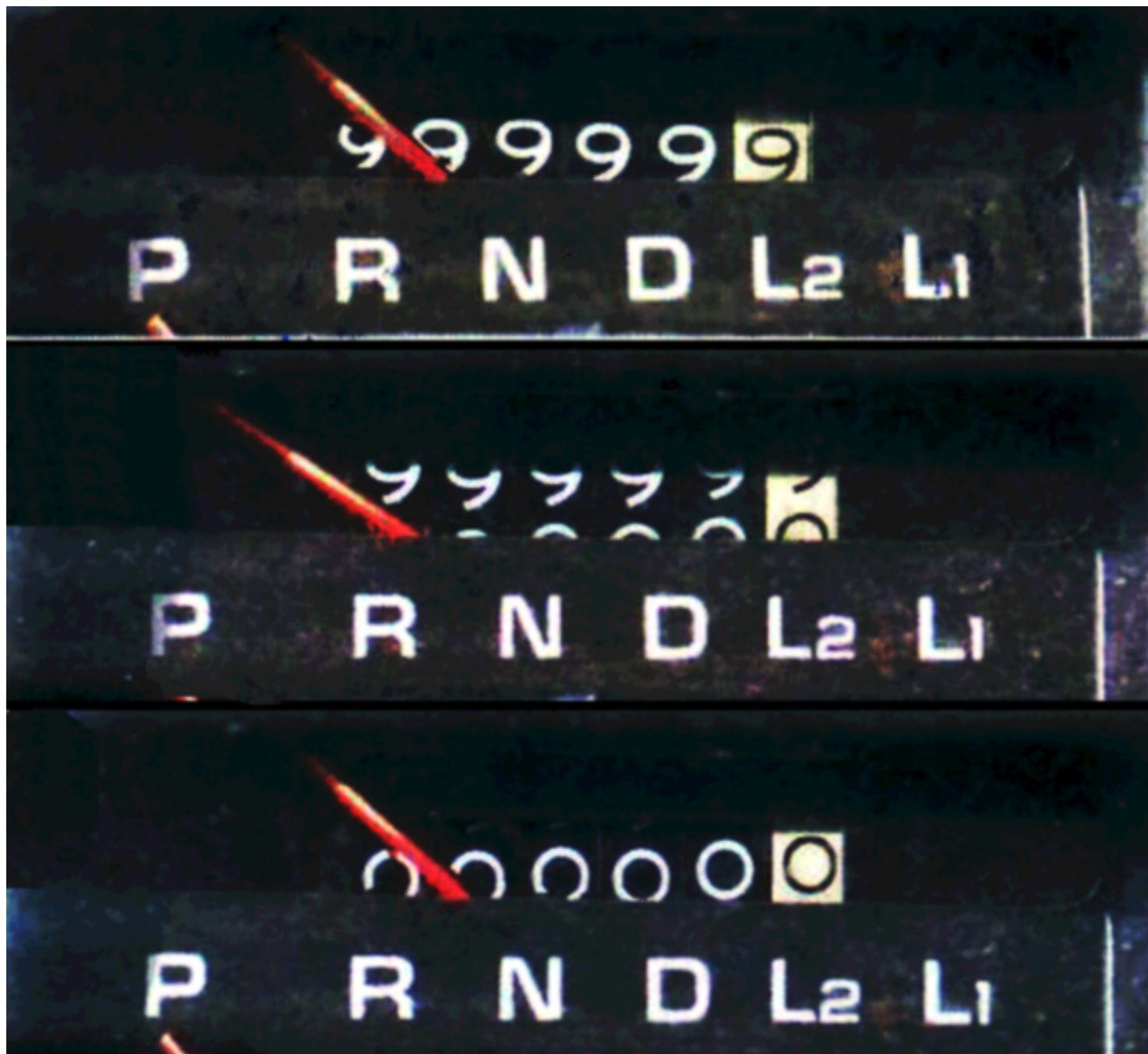
# Syntax jazyka C - dátové typy

```
#include <stdio.h>

int main() {
    int a = 2147483650;
    printf("%d", a);
}
```



# Syntax jazyka C - dátové typy



# Syntax jazyka C - Ascii

## ASCII Hex Symbol

48	30	0
49	31	1
50	32	2
51	33	3
52	34	4
53	35	5
54	36	6
55	37	7
56	38	8
57	39	9
58	3A	:
59	3B	;
60	3C	<
61	3D	=
62	3E	>
63	3F	?

## ASCII Hex Symbol

64	40	@
65	41	A
66	42	B
67	43	C
68	44	D
69	45	E
70	46	F
71	47	G
72	48	H
73	49	I
74	4A	J
75	4B	K
76	4C	L
77	4D	M
78	4E	N
79	4F	O

## ASCII Hex Symbol

80	50	P
81	51	Q
82	52	R
83	53	S
84	54	T
85	55	U
86	56	V
87	57	W
88	58	X
89	59	Y
90	5A	Z
91	5B	[
92	5C	\
93	5D	]
94	5E	^
95	5F	_

## ASCII Hex Symbol

96	60	`
97	61	a
98	62	b
99	63	c
100	64	d
101	65	e
102	66	f
103	67	g
104	68	h
105	69	i
106	6A	j
107	6B	k
108	6C	l
109	6D	m
110	6E	n
111	6F	o

## ASCII Hex Symbol

112	70	p
113	71	q
114	72	r
115	73	s
116	74	t
117	75	u
118	76	v
119	77	w
120	78	x
121	79	y
122	7A	z
123	7B	{
124	7C	
125	7D	}
126	7E	~
127	7F	



# Syntax jazyka C - Ascii

```
#include <stdio.h>
```

```
int main() {  
    char a = 'a';  
    char b = 'b';  
    char z = 'z';  
    char Z = 'Z';  
  
    printf("a je %d \n", a);  
    printf("b je %d \n", b);  
    printf("z je %d \n", z);  
    printf("Z je %c \n", Z);  
}
```

ASCII Hex Symbol

64	40	@
65	41	A
66	42	B
67	43	C
68	44	D
69	45	E
70	46	F
71	47	G
72	48	H
73	49	I
74	4A	J
75	4B	K
76	4C	L
77	4D	M
78	4E	N
79	4F	O

ASCII Hex Symbol

80	50	P
81	51	Q
82	52	R
83	53	S
84	54	T
85	55	U
86	56	V
87	57	W
88	58	X
89	59	Y
90	5A	Z
91	5B	[
92	5C	\
93	5D	]
94	5E	^
95	5F	_

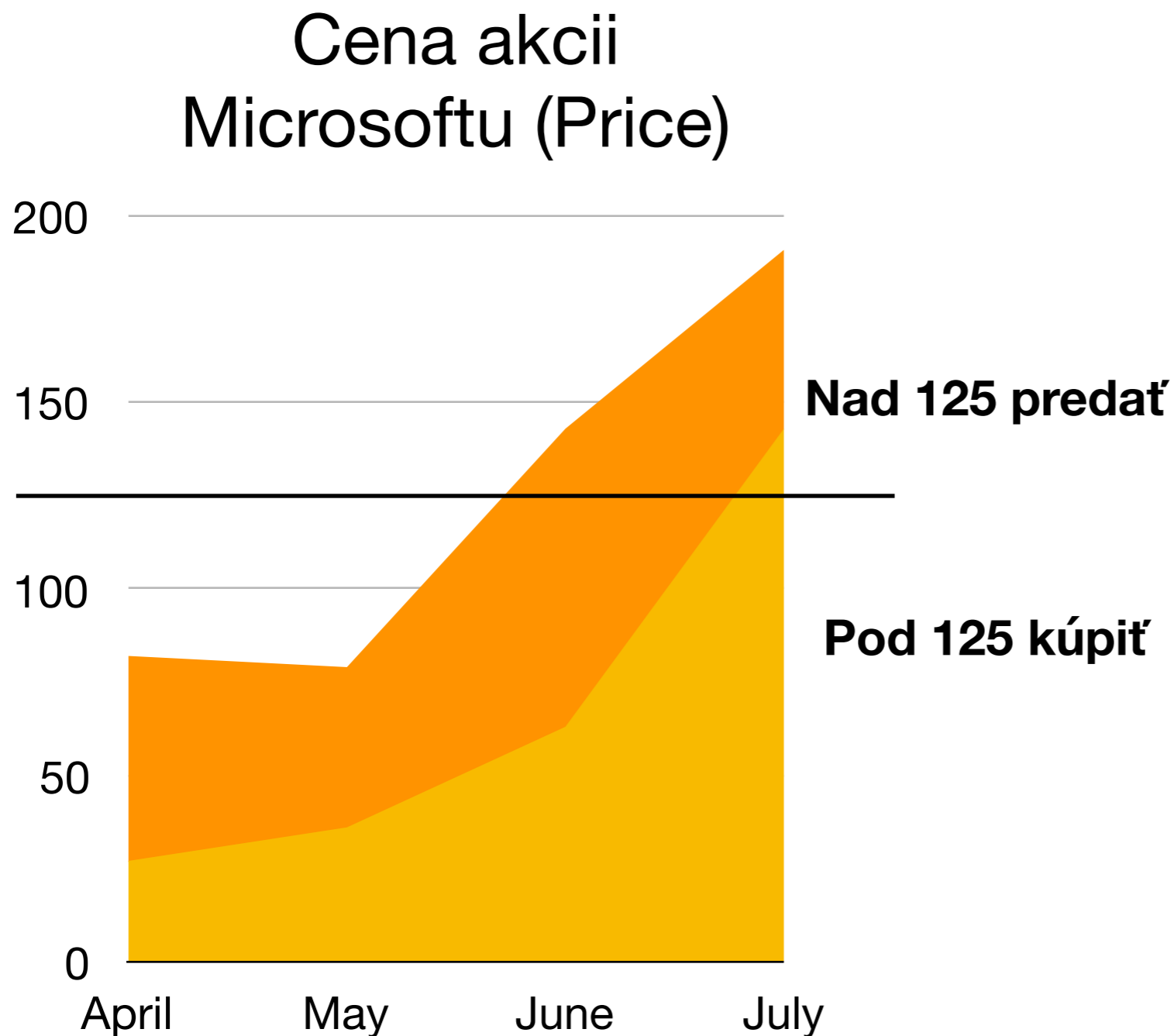
ASCII Hex Symbol

96	60	`
97	61	a
98	62	b
99	63	c
100	64	d
101	65	e
102	66	f
103	67	g
104	68	h
105	69	i
106	6A	j
107	6B	k
108	6C	l
109	6D	m
110	6E	n
111	6F	o

ASCII Hex Symbol

112	70	p
113	71	q
114	72	r
115	73	s
116	74	t
117	75	u
118	76	v
119	77	w
120	78	x
121	79	y
122	7A	z
123	7B	{
124	7C	
125	7D	}
126	7E	~
127	7F	

# If-else statement (vetvenie)



Kúpiť alebo predat?



# If-else statement (vetvenie)

```
#include <stdio.h>

int main() {
    int price;
    printf("Enter the current price of MSFT: ");
    scanf("%d", &price);

    // if price is over 125 then sell otherwise buy
    if(price > 125) {
        printf("SELL ! SELL !");
    }
    else{
        printf("BUY !!!");
    }
}
```



Kúpiť alebo predat?

# If-else statement (vetvenie)

```
#include <stdio.h>

int main() {
    int price;
    printf("Enter the current price of MSFT:
");
    scanf("%d", &price);

    // if price is over 125 then sell otherwise
buy
    if(price > 125) {
        printf("SELL ! SELL !");
    }
    else if (price == 125){
        printf("Watch TV and rest");
    }
    else {
        printf("BUY !!!");
    }
}
```




Kúpiť alebo predat?

# while statements (“pokial” loop)

```
#include <stdio.h>
```

```
int main() {  
    int price;  
    printf("\n Zadaj pociatocnu cenu akcie : ");  
    scanf("%d", &price);  
  
    while(price < 200) {  
        printf("Nakupil si: 1 akciu MSFT \n Zadaj aktualnu cenu akcie : \n");  
        scanf("%d", &price);  
    }  
  
    printf("Cena je viac ako 200$, uz si nechcel nakupovat");  
}
```



Chcem nakupovať akcie len keď  
sú pod cenou 200 \$

# while statements (“pokiał” loop)

```
while (podmienka - Pravda alebo Nepravda) {  
    Urob nieco;  
}
```

# if/else vs while loop

```
int a = 1;
```

```
if (a==1) {
```

```
    // do something, for example print something
```

```
    printf("something");
```

```
}
```

```
while (a==1) {
```

```
    printf("something");
```

```
}
```

Čo sa stane v tomto prípade?

Pokiaľ je pravdivé tak rob

# Function (funkcie)

**Pomenovaný blok kódu, ktorý vykoná nejakú úlohu a vráti kontrolu späť hlavnej funkcii**





# Function (funkcie)

**Pomenovaný blok kódu, ktorý vykoná nejakú úlohu a vráti kontrolu späť hlavnej funkcii**



# Function (funkcie)

Pomenovaný blok kódu, ktorý vykoná nejakú úlohu a vráti kontrolu späť hlavnej funkcii



main function()

Vytvorím funkciu zavolám ju a po získaní hodnoty pokračujem v práci!

```
int getMax(int a, int b) {  
    if (a>b) {  
        return a;  
    }  
    else {  
        return b;  
    }  
}  
  
int main() {  
    int a = 2;  
    int b = 4;  
    int max = getMax(a, b);  
    printf("%d", max);  
}
```

# Function syntax

Návratová hodnota    Názov parametru

```
int nazovFunkcie(int parameter1, int parameter2) {
```

```
    ...  
    ...  
    return 5;  
}
```

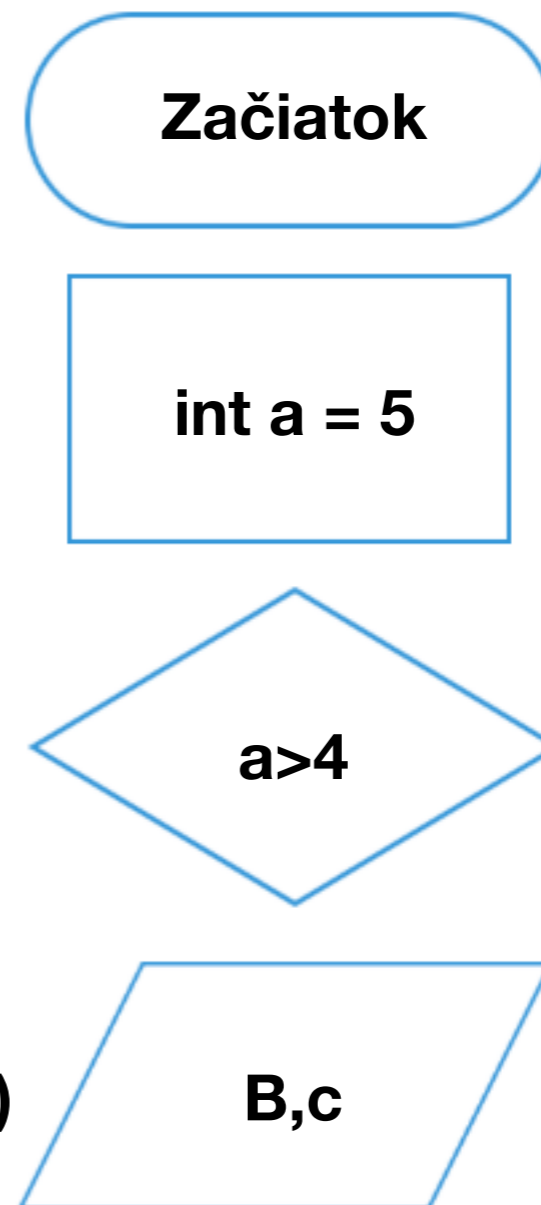
Typ parametru

Navrátená hodnota hlavnému programu



# Vývojové diagramy - úvod

- **Začiatok alebo koncový bod procesu**
- **Výraz (Statement), priradenie, printf ...**
- **Symbol “if/else” alebo “while akcie”**
- **Symbol pre vstup/výstup v programe (scanf)**



# Záver

- **Rekapitulovanie dátových typov (int, char, float, double)**
- **Naprogramovanie základných algoritmov**
- **Ukážka vývojových diagramov a spôsobu ich tvorenia**

**Ďakujem za pozornosť**

# Úloha č.1

**Ciel':** Vypočítať faktoriál čísla

**Vstup:** Celé číslo



**Postup:**

- Vytvoriť funkciu `factorial(int n)`, ktorá vráti faktoriál čísla `n`
- Príklad: zadám číslo  $5! = 5*4*3*2*1 = 120$  - príklad výpisu
- **Ošetriť, že program vráti chybovú správu -1 a text, ak sa zadá ne celé číslo**

# Úloha č.2

**Ciel':** Vypočítať všetkých deliteľov čísla  $n$  a  $m$

**Vstup:** Celé číslo  $m$  a  $n$

**Postup:**

- Vypísať všetkých deliteľov  $m$  a  $n$
- V prípade, ak nemá deliteľa je to prvočíslo (vypísať)
- Nájsť najväčšieho spoločného deliteľa  $n$  a  $m$





# Úloha č.3

**Ciel': Naprogramovať simuláciu kupovania / predávania akcií**

**Vstup: počet kôl, koľko krát má bežať hra**

**Postup:**

- **Zadať číslo od 1-100 koľko krát sa má program spustiť**
- **Vygenerovať náhodné číslo od 0-100**
- **Pokiaľ náhodné číslo je  $> 50$  tak predáť akciu**
- **Pokiaľ náhodné číslo je  $< 50$  kúpiť akciu**
- **Pokiaľ náhodné číslo je  $= 50$  tak nerobiť nič**
- **Na konci systém vypíše, koľko máte peňazí a akcií**



**Pokračovanie: V poslednom “kole” simulácie predáť všetky akcie za vygenerovanú náhodnú cenu a zistiť stav účtu**

**Pokračovanie II: Overiť, že ak nemáte peniaze tak sa program ukončí resp. Nerobí nič**