**PRÍPRAVA NA VYUČOVACIU HODINU**

**INFORMATIKA Príprava*\_VH\_INF\_DID2\_PIS\_PASCAL***

|  |  |
| --- | --- |
| **Vyučujúci:** | Ing. Anton Pisko |
| **Dátum:** |  |
| **Trieda:** | 3. ročník gymnázia III.A |
| **Predmet:** | Informatika |
| **Tematický celok:** | Programovanie - podmienené príkazy |
| **Téma:** | Neúplný podmienený príkaz, užívateľom definované typy. |
| **Cieľ/ ciele vyučovacej hodiny:** | 1.Názorne popísať a vysvetliť použitie jednotlivých typov podmienených príkazov.  2.Dokázať zostaviť jednoduchý program a správne  použiť neúplného podmieneného príkazu v programovacom prostredí jazyka Pascal.  3.Správne algoritmicky zoradiť a doplniť podmienený príkaz podľa pracovného listu.  4.Upevniť získané vedomosti a zdokonaliť zručnosti pri programovaní  v jazyku Pascal. |
| **Typ vyučovacej hodiny:** | Vyučovacia hodina základného typu: 2 x 45 min. |
| **Vyučovacie metódy:** | * Diskusia * Frontálna práca žiakov * Individuálna práca * Prezentovanie výstupov |
| **Medzipredmetové vzťahy:** | Matematika, Finančná gramotnosť |

|  |  |
| --- | --- |
| **Pomôcky:** | Interaktívna tabuľa, dataprojektor, samostatný počítač pre každého žiaka s programovým vybavením, pripojenie na Internet, pracovný list  Spracované podporné materiály: [**Pascal\_podmieneny\_prikaz (ppt)**](file:///C:\Users\EVSINF1\AppData\Local\Temp\Pascal_podmieneny_prikaz.pptx)  [**Kartičky - podmienky**](file:///C:\Users\EVSINF1\AppData\Local\Temp\podmienka.exe)  [**prac\_list\_podmienený príkaz**](file:///C:\Users\EVSINF1\AppData\Local\Temp\praclist_did2.docx) |

|  |  |  |
| --- | --- | --- |
| **čas:** | **činnosť učiteľa:** | **činnosť žiakov:** |
| 2 min.  3 min.    5 min.    12 min.      3 min.      15 min.    5 min. | 1. Administratívne záležitosti VH, zápis do TK, oboznámenie žiakov s témou a so štruktúrou vyučovacej hodiny. 2. a, Premietnutie ukážky funkčného programu s neúplným podmieneným príkazom pomocou dataprojektora a interaktívnej tabule   b, Učiteľ môže predviesť funkčné zdrojové kódy s použitím neúplného podmieneného príkazu.   1. Otázkou: „Dokázali by ste vytvoriť   a  uviesť príklad na jednoduchý program s využitím neúplného podmieneného príkazu?“ -navodenie diskusie o využití jednotlivých možností podmienok v programovaní.  4. Výklad učiva podľa materiálu: **výklad\_podmienky** učiteľ kontrolnými otázkami overuje či žiaci porozumeli výkladu učiva, tu dáva dôraz na vysvetlenie a použitie príkladov v prostredí programu PASCAL pričom príklady aj predvedie (čísla).  Na aktivizáciu žiakov, pred zadaním úlohy učiteľ predvedie animovanú skladačku v programe Kartičky: [**podmienky**](file:///C:\Users\EVSINF1\AppData\Local\Temp\podmienka.exe)  ,vyzve žiakov: “Kto si vyskúša algoritmicky správne zoradiť jednotlivé kroky zdrojoveho kódu? “  5. Zadanie úlohy – spracujte podľa predloženého prac. listu zdrojový kód a overte jeho funkčnosť **praclist\_podmienky**  Táto úloha nadväzuje na úlohu:  **Vytvorte program na riešenie úlohy o výplate peňazí, v ktorom treba určiť, koľko 500, 200, 100, 50, 20, 10, 5, 2, 1 - eur potrebujeme na vyplatenie danej sumy tak, aby sme použili najmenší počet platidiel.**  6. Výzva na predvedenie úlohy.  7. Učiteľ kontroluje splnenie zadania úlohy a vyhodnotí úroveň jej spracovania. Úlohy ohodnotí, pochváli. Niektoré môže oznámkovať. Vyhodnotenie splnenia cieľov vyučovania, zadanie námetu na domácu úlohu, ukončenie VH. | 1. Žiaci sledujú učiteľa 2. Žiaci sledujú ukážky   Učiteľ vyvolá diskusiu na tému,kedy sa musíme rozhodovať na základe podmienok, žiaci uvádzajú konkrétne príklady zo života   1. Sledujú výklad učiva a odpovedajú na kontrolné otázky učiteľa. 2. Žiaci sledujú výklad učiva a robia si poznámky   Na interaktívnej tabuli  skúšajú správne zoradiť skladačku -podmienky   1. Pracujú na zadaní úlohy,   referujú  o stave vypracovania  úlohy, resp. úloh.     1. Spracované úlohy si žiaci uložia   v počítači pod názvom vyplata.pas  spoločne s učiteľom vyhodnotia  postupy pri riešení algoritmizácie  danej úlohy. |
| **Popis edukačného programu:**  **Motivácia**:   1. Ukážka funkčného programu s použitím podmieneného príkazu. 2. Diskusia o využití podmienok na základe rozhodovania sa v rôznych oblastiach života.   **Expozícia**:   1. Premietnutie iných postupov riešenia zdrojových kódov a ich porovnanie s riešením pomocou podmieneného príkazu (viz.) 2. Výklad učiva: podmienené príkazy - neúplný podmienený príkaz v programovacom jazyku PASCAL užívateľom definované typy 3. Frontálna práca žiakov na interaktívnej tabuli ( Animovaná skladačka -PODMIENKY). 4. Vypracovanie úlohy podľa zadania v pracovnom liste   **Fixácia**:   1. Samostatné vypracovanie úlohy podľa zadania a následná prezentácia (predvedenie) vypracovanej úlohy. 2. Zadanie kontrolných otázok. 3. Zhrnutie poznatkov, vyhodnotenie, zadanie domácej úlohy. | | |

V Gelnici, dňa :

Vypracoval: Ing. Anton Pisko ......................................

(meno a priezvisko) (podpis)

**Príprava vyučovacej hodiny – výklad učiva**

# Téma: Neúplný podmienený príkaz.

**Úloha pre expozíciu a fixáciu učiva je zostrojenie a predvedenie funkčného zdrojového kódu : Vytvorte program na riešenie úlohy o výplate peňazí, v ktorom treba určiť, koľko 500, 200, 100, 50, 20, 10, 5, 2, 1 - euroviek potrebujeme na vyplatenie danej sumy tak, aby sme použili čo najmenší možný počet platidiel.**

Predpokladané vstupné požiadavky na vedomosti, zručnosti a návyky:

Žiak pozná prostredie programovacieho jazyka Pascal, dokáže pracovať so základnými príkazmi vstupu a výstupu, priraďovacím príkazom a so zloženými príkazmi, orientuje sa v štruktúre programu, rozlišuje jednotlivé dátové typy a dokáže ich správne deklarovať, používa logické operátory, pozná cyklus pre určitý počet opakovaní – for, pozná prácu s prvkami poľa.

**Nové učivo:**

[**Prezentácia NEÚPLNÝ PODMIENENÝ PRÍKAZ je samostatnou súčasťou tejto prípravy**](file:///C:\Users\EVSINF1\AppData\Local\Temp\Pascal_podmieneny_prikaz.pptx)

**Príkazy vetvenia**

Príkaz vetvenia je zložený riadiaci príkaz. Používame ho vtedy, keď potrebujeme, aby sa určitý príkaz alebo príkazy vykonal/vykonali iba vtedy, keď je splnená nejaká podmienka. V programe sa táto vykoná iba raz!

Príkazy vetvenia sú:

**a)   preskok** – neúplný podmienený príkazu vetvenia, testuje logickú podmienku – **if** bez else

**b)   alternujúce vetvenie** – úplný podmienenýo príkaz vetvenia, testuje logickú podmienku – **if** s else

**c)   prepínač** – testuje hodnotu premennej, alebo funkcie, má aspoň tri vetvy – **case**

**Preskok -** neúplný podmienený príkaz vetvenia **if**

|  |  |
| --- | --- |
| Štruktúrogram:  http://www.programovanie.kromsat.sk/prog-b/s17/img004.gif | Zápis v Pascale:  **if *výraz* then *príkaz1*;** |

 Ako to funguje? Najprv sa vyhodnotí logický výraz. Ak je pravdivý, vykoná sa príkaz1 a potom sa pokračuje za príkazom vetvenia. Ak je nepravdivý, príkaz1 sa preskočí a pokračuje sa za príkazom vetvenia.Na vysvetlenie funkcie neúplného podmieneného príkazu vetvenia ( preskoku) učiteľ predvedie príklady,pričom položí dôraz na vysvetlenie syntaxe zápisu a jeho začlenenie v  programe.

Príklad 1: program načíta celé číslo a vypíše jeho absolútnu hodnotu.

program abs;

uses crt;

var x: integer;

begin

writeln('Zadaj cele cislo');

**if x<0 then x:= x\*(-1);**

readln(x);

Syntax zápisu neúplného podmieneného príkazu vetvenia v zdrojovom kóde

Príklady programov sú súčasťou prezentácie

**if x<0 then x:= x\*(-1);**

writeln ('Absolutna hodnota zadaneho cisla je ', x);

readln;

end.

Príklad 2: program vypíše číslo zadané z klávesnice, len ak je menšie ako 10.

program podmienka;

var

c: integer;

begin

**if c < 10 then**

writeln('zadajte cele cislo');

readln(c);

**if c < 10 then**

writeln('cislo ', c, ' je mensie ako 10');

readln;

end. tento príklad bude v [( Animovaná skladačka – PODMIENKY )](file:///C:\Users\EVSINF1\AppData\Local\Temp\podmienka.exe)

|  |  |  |
| --- | --- | --- |
| Žiaci si zároveň skúšajú zápis neúplného podmieneného príkazu vetvenia do už rozpracovaného zdrojového kódu z predošlej vyučovacej hodiny ( každý žiak ho má v priečinku „ZDROJOVE\_KODY“ uložený) a experimentujú so zmenou výrazu pre podmienku ( skúšajú meniť **Logické operátory porovnávania** | | |
| = | rovná sa |
| < | je menší ako |
| <= | je menší alebo rovný |
| > | je väčší ako |
| >= | je väčší alebo rovný |
| <> | nerovná sa a sledujú ako sa správa program |

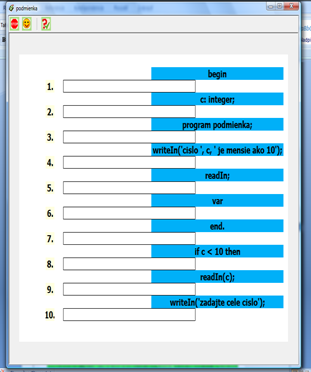
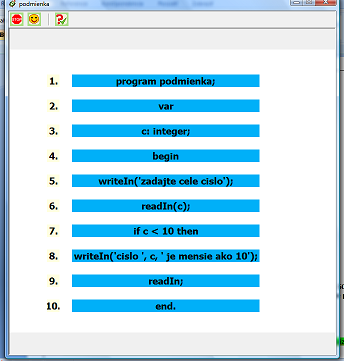
Učiteľ sa pýta, či žiaci porozumeli výkladu učiva a kontrolnými otázkami si overuje:

Čo sa stane, ak:

Zmeníte logické operátory porovnávania ?

Dokázali by ste zapísať ďalší neúplný podmienený príkaz vetvenia pre porovnanie dvoch rôznych čísel ?

Pre motiváciu na riešenie úlohy učiteľ vyzve na frontálnu prácu žiakov na interaktívnej tabuli [(Animovaná skladačka – PODMIENKY](file:///C:\Users\EVSINF1\AppData\Local\Temp\podmienka.exe), spracovaná v programe Kartičky)

**[](file:///C:\Users\EVSINF1\AppData\Local\Temp\podmienka.exe)**

Animovaná skladačka – PODMIENKY pred a po zložení

Po výklade učiva nasleduje riešenie úlohy ( túto riešia žiaci spoločne s učiteľom):

**Vytvorte program na riešenie úlohy o výplate peňazí, v ktorom treba určiť, koľko 500, 200, 100, 50, 20, 10, 5, 2, 1 - eur potrebujeme na vyplatenie danej sumy tak, aby sme použili najmenší počet platidiel.**

Tu učiteľ formou ukážky predvedie, ako je riešená táto úloha bez použitia podmienených príkazov

(považujem za dôležité, aby si žiaci aj na základe uvedených príkladov uvedomili, výhodnosť

riešenia úlohy viacerými spôsobmi a vedeli tak porovnať, ktoré z riešení je z hľadiska algoritmizácie pre danú úlohu najefektívnejšie, rôznymi postupmi sa dopracujem k rovnakému výsledku )

1. **ukážka riešenia ( zložité na zápis )**

program vyplaty\_pre\_zamestnancov;

var pocet:word;

plat:longint;

i:word;

b500,b200,b100,b50,b20,b10,b5,b2,b1:word;

c500,c200,c100,c50,c20,c10,c5,c2,c1:word;

begin

c500:=0; c200:=0; c100:=0; c50:=0; c20:=0; c10:=0; c5:=0; c2:=0; c1:=0;

b500:=0; b200:=0; b100:=0; b50:=0; b20:=0; b10:=0; b5:=0; b2:=0; b1:=0;

writeln('Zadaj pocet zamestnancov ktori dostanu vyplatu');

readln(pocet);

for i:=1 to pocet do begin

writeln('Zadaj plat ',i,'. zamestnanca');

readln(plat);

b500:=plat div 500; { kolko 500-eur naviac }

plat:=plat mod 500; { co ostane }

b200:=plat div 200; { kolko 200-eur naviac }

plat:=plat mod 200; { co ostane }

b100:=plat div 100; { kolko 100-eur naviac }

plat:=plat mod 100; { co ostane }

b50:=plat div 50; { kolko 50-eur naviac }

plat:=plat mod 50; { co ostane }

b20:=plat div 20; { kolko 20-eur naviac }

plat:=plat mod 20; { co ostane }

b10:=plat div 10; { kolko 10-eur naviac }

plat:=plat mod 10; { co ostane }

b5:=plat div 5; { kolko 5-eur naviac }

plat:=plat mod 5; { co ostane }

b2:=plat div 2; { kolko 2-eur naviac }

plat:=plat mod 2; { co ostane }

b1:=plat div 1; { kolko 1-eur naviac }

plat:=plat mod 1; { co ostane }

c500:=c500+b500; c200:=c200+b200; c100:=c100+b100;

c50:=c50+b50; c20:=c20+b20; c10:=c10+b10;

c5:=c5+b5; c2:=c2+b2; c1:=c1+b1;

end;

writeln('Uctovnik potrebuje:');

writeln( c500,' 500-eur bankoviek');

writeln( c200,' 200-eur bankoviek');

writeln( c100,' 100-eur bankoviek');

writeln( c50,' 50-eur bankoviek');

writeln( c20,' 20-eur bankoviek');

writeln( c10,' 10-eur bankoviek');

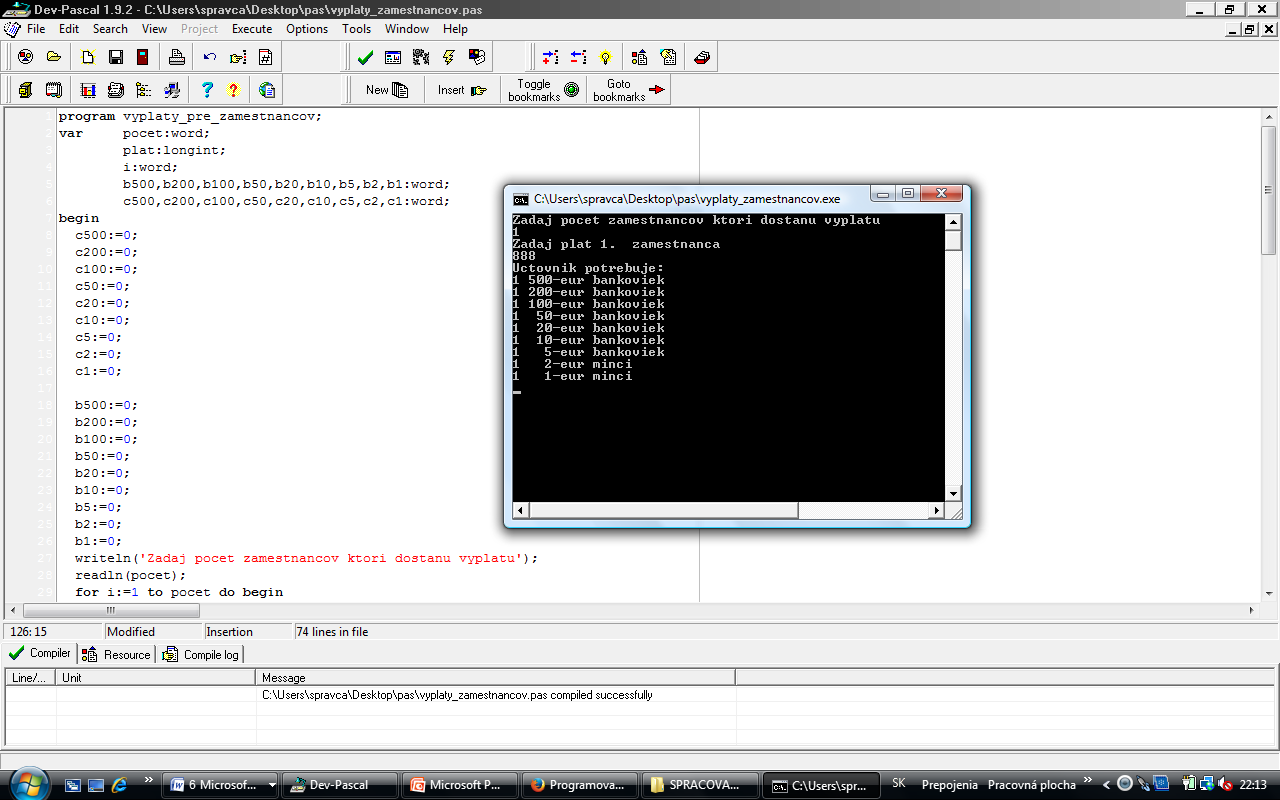
writeln( c5,' 5-eur bankoviek');

writeln( c2,' 2-eur minci');

writeln( c1,' 1-eur minci');

readln;

end.

****

náhľad na hotový program 1. Riešenie

1. **ukážka riešenia (pomocou procedúry )**

Program platidla;