

Řešené úlohy z oblasti zpracování informací: algoritmizace úloh

Bc. Martin Majer

Diplomová práce
2021

 Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav informatiky a umělé inteligence

Akademický rok: 2020/2021

ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: Bc. Martin Majer
Osobní číslo: A19698
Studijní program: N3902 Inženýrská informatika
Studijní obor: Učitelství informatiky pro střední školy
Forma studia: Prezenční
Téma práce: Řešené úlohy z oblasti zpracování informací: algoritmizace úloh
Téma práce anglicky: Resolved Tasks in the Information Processing Field: The Algorithmisation of Tasks

Zásady pro vypracování

1. Seznamte se s rámcovým vzdělávacím programem pro střední školy.
2. Provedte analýzu používaného softwaru při výuce algoritmizace úloh na středních školách.
3. Popište využívaný software při výuce algoritmizace úloh používaný na gymnáziích a středních školách.
4. Vytvořte sadu podkladů pro výuku včetně pracovních listů.
5. Prakticky ověřte sadu úloh pomocí dotazníku.

Forma zpracování diplomové práce: **Tištěná/elektronická**

Seznam doporučené literatury:

- PECINOVSKÝ, Rudolf. *Začínáme programovat v jazyku Python*, Grada, 2020, ISBN 978-80-271-1237-1.
PECINOVSKÝ, Rudolf. *Python – kompletní příručka jazyka pro verzi 3.9*, Grada, 2020, ISBN 978-80-271-1269-2.
HYLMAR, Radek. *Programování pro úplné začátečníky*, Computer Press, 2019, ISBN 978-80-251-2129-0.
PŠENČÍKOVÁ, Jana. *Algoritmizace*, Computer Media, 2016, ISBN 978-80-7402-034-6.
LANDA, Lev Nachmanovič. *Algoritmy a učení: Kybernetika, algoritmizace a heuristika ve vyučování*, Státní pedagogické nakladatelství Praha, 1973.
VYSTAVĚL, Radek. *Moderní programování – učebnice pro středně pokročilé*, moderní Programování 2013, ISBN 978-80-903951-7-6.
KMOCH, Petr. *Informatika a výpočetní technika pro střední školy*, Computer Press, 2004, ISBN 978-80-251-0376-0.
ROUBAL, Pavel. *Informatika a výpočetní technika pro střední školy*. Computer Press, 2017, ISBN 978-80-251-3227-2.

Vedoucí diplomové práce: **Ing. Karel Perůtka, Ph.D.**
Ústav řízení procesů

Datum zadání diplomové práce: **15. ledna 2021**

Termín odevzdání diplomové práce: **17. května 2021**

doc. Mgr. Milan Adámek, Ph.D. v.r.
děkan



prof. Mgr. Roman Jašek, Ph.D. v.r.
ředitel ústavu

Ve Zlíně dne 15. ledna 2021

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně;
- byl/a jsem seznámen/a s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 5. 5. 2021

Bc. Martin Majer, v. r.

ABSTRAKT

Diplomová práce se zabývá tvorbou výukových materiálů na téma: řešené úlohy z oblasti zpracování informací: algoritmizace úloh. Konkrétně jde o část algoritmizace a o část výuky základů programování. Materiály jsou tvořeny na úrovni středních škol a gymnázií. Jako programovací jazyk byl na základě provedeného průzkumu vybrán jazyk Python. Výukové materiály obsahují průvodní dokument, kde je popsáno, co a jak učit, výukové podpůrné prezentace, sadu testů pro oba vyučované okruhy a také návod pro práci s používanými softwary, konkrétně jde o software IDLE pro výuku programovacího jazyku Python. V závěrečné části výukové materiály ohodnotí za pomoci dotazníku učitelé ze středních škol.

Klíčová slova: vyučování, výukové materiály, algoritmizace, programování, Python, IDLE

ABSTRACT

The diploma thesis deals with the creation of teaching materials on the topic: solved problems in the field of information processing: algorithmization of tasks, specifically it is part of algorithmization and part of teaching the basics of programming. The materials are created at the level of high schools and grammar schools. As a programming language, Python was selected based on the survey. The teaching materials include an accompanying document describing what and how to teach, teaching support presentations, a set of tests for both areas taught, as well as instructions for working with the software used, specifically IDLE software for teaching the Python programming language. In the final part, teaching materials will be evaluated, with the help of a questionnaire, by teachers from secondary schools.

Keywords: teaching, electronic textbooks, algorithmization, programming, Python (programming language), IDLE

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Karlu Perůtkovi, Ph.D. za ochotu, vstřícnost a cenné rady pro zpracování této diplomové práce. Dále také mojí rodině za poskytnuté zázemí a trpělivost, jak při vypracovávání diplomové práce, tak po celou dobu studia.

OBSAH

ÚVOD.....	7
I TEORETICKÁ ČÁST.....	8
1 RÁMCOVÝ VZDĚLÁVACÍ PROGRAM.....	9
1.1 RVP PRO STŘEDNÍ ŠKOLY.....	9
1.1.1 RVP pro gymnázia.....	9
1.1.2 RVP pro obor Informační technologie.....	10
1.2 INFORMATICKÉ MYŠLENÍ.....	11
2 ANALÝZA VYUŽÍVANÉHO SOFTWARE PRO VÝUKU PROGRAMOVÁNÍ NA STŘEDNÍCH ŠKOLÁCH.....	12
2.1 PROVÁDĚNÍ VÝZKUMU.....	12
2.2 VYHODNOCENÍ VÝZKUMU.....	13
3 POPIS POUŽÍVANÉHO SOFTWARE.....	14
3.1 PYTHON.....	14
3.1.1 Historie jazyku Python.....	15
3.1.2 Výhody a nevýhody jazyku Python.....	15
3.1.2.1 Výhody.....	16
3.1.2.2 Nevýhody.....	16
3.2 IDLE.....	16
3.2.1 Výhody a nevýhody prostředí IDLE.....	17
3.2.1.1 Výhody.....	17
3.2.1.2 Nevýhody.....	17
3.2.2 Práce v IDLE.....	17
4 TVORBA VÝUKOVÝCH MATERIÁLŮ.....	18
4.1 ALGORITMIZACE.....	18
4.2 PROGRAMOVÁNÍ.....	19
4.3 TVORBA SAMOTNÝCH VÝUKOVÝCH MATERIÁLŮ.....	19
II PRAKTICKÁ ČÁST.....	20
5 PODKLADY PRO VÝUKU.....	21
6 DOTAZNÍKOVÉ OVĚŘENÍ VÝUKOVÝCH MATERIÁLŮ.....	78
6.1 VYHODNOCENÍ DOTAZNÍKU A ZPĚTNÉ VAZBY.....	83
ZÁVĚR.....	85
SEZNAM POUŽITÉ LITERATURY.....	86
SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....	88
SEZNAM OBRÁZKŮ.....	89
SEZNAM TABULEK.....	90
SEZNAM PŘÍLOH.....	91

ÚVOD

Diplomovou prací tvoří výukové materiály na téma řešené úlohy z oblasti zpracování informací. Jedním z důvodů, proč jsem si vybral konkrétní téma, je, že se chci věnovat výuce informatiky na střední škole či gymnáziu, tedy stát se učitelem. Téma algoritmizace a programování je jedno ze základních témat, která tvoří kostru výuky informatiky na středních školách potažmo gymnáziích. Základní téma se dělí na část výuky algoritmizace a na část výuky základů programování.

Před samotnou výukou je nutné provést analýzu RVP pro střední školy a gymnázia, pro výuku informatiky na gymnáziích i odborných středních školách.

Dále bude provedeno dotazníkové šetření o tom, jaký programovací jazyk je v reálném světě a středních školách a gymnáziích nejpoužívanějších. Dle situace v profesní sféře lze očekávat, že se bude jednat o jazyk Python.

Následně bude vybráno vývojové prostředí pro vyučovaný programovací jazyk. Bude kladen důraz na jednoduchost instalace a na snadné ovládání. Pokud bude podle předpokladů zvolen za programovací jazyk Python, jako prostředí bude zvoleno IDLE, protože je po instalaci samotného Pythonu nainstalováno taky a je jednoduché pro začátečníky.

Poté bude vytvořena sada podkladů pro výuku. Nejprve bude nutno navrhnout, s kolika hodinami celkem se bude počítat. Posléze bude vytvořen průvodní dokument, kde budou uváděny, všechny obecné náležitosti jako cíle, klíčové kompetence atd. Taktéž bude v tomto dokumentu podrobně popsána každá jednotlivá vyučovací hodina s časovým rozpisem, komentářem pro vyučujícího a se vzorovými příklady.

V rámci výukových materiálů hrají důležitou roli prezentace sloužící jako podklad k odborným tématům. Nedílnou součástí budou také testy pro ověření žáky nabytých znalostí a návody pro práci v jednotlivých programech.

Na závěr budou tyto vytvořené materiály otestovány pomocí dotazníku, kde respondenti budou učitelé ze středních škol a gymnázií, čili učitelé z praxe. Dopředu lze ale očekávat, že komplexní zpětná vazba bude od učitelů z praxe celkem těžko k sehnání, jelikož to bude pro ně značná časová zátěž.

I. TEORETICKÁ ČÁST

1 RÁMCOVÝ VZDĚLÁVACÍ PROGRAM

„Rámcové vzdělávací programy (RVP) tvoří obecně závazný rámec pro tvorbu školních vzdělávacích programů škol všech oborů vzdělání v předškolním, základním, základním uměleckém, jazykovém a středním vzdělávání. Do vzdělávání v České republice byly zavedeny zákonem č. 561/2004 Sb., o předškolním, základním, středním, vyšším odborném a jiném vzdělávání (školský zákon).“ [1]

1.1 RVP pro střední školy

V mé diplomové práci jsem se rozhodl, že budu vycházet z rámcových vzdělávacích programů pro gymnázia a z rámcového vzdělávacího programu pro obor Informační technologie, jelikož s těmito obory mám zkušenost a myslím si, že zde je místo pro výuku programování. Značná část oslovených středních škol vyučovala podle jednoho z těchto RVP.

Jelikož platné RVP již lehce zastarávají, v nejbližší budoucnosti budou zrevidovány, aby lépe odpovídaly požadavkům dnešní doby. Výuka algoritmizace bude i do budoucna stále důležitou částí výuky.

1.1.1 RVP pro gymnázia

V příslušném RVP jsem našel kapitolu s názvem: Informatika a informační a komunikační technologie, která dopodrobna rozebírá téma informatiky na gymnáziích, zvláště bych vyvedl následující odstavec:

„V rámci oblasti Informatika a ICT se žák seznámí se základy informatiky jako vědního oboru, který studuje výpočetní a informační procesy z hlediska používaného hardwaru i softwaru, a s jejím postavením v moderním světě. Cílem je zpřístupnit žákům základní pojmy a metody informatiky, napomáhat rozvoji abstraktního, systémového myšlení, podporovat schopnost vhodně vyjadřovat své myšlenky, smysluplnou argumentací je obhajovat a tvůrčím způsobem přistupovat k řešení problémů. Žák se seznámí se základními principy fungování prostředků ICT a soustředí se na pochopení podstaty a průběhu informačních procesů, algoritmického přístupu k řešení úloh a významu informačních systémů ve společnosti.“ [2]

V souladu s RVP pro gymnázia a výše uvedeného odstavce se budu ve své práci zabývat cílem soustředění se na algoritmické myšlení.

Toto mé tvrzení se nachází i v konkrétním vyjmenování cílů zaměření vzdělávací oblasti:

„uplatňování algoritmického způsobu myšlení při řešení problémových úloh“ [2]

1.1.2 RVP pro obor Informační technologie

Toto RVP je jinak stylizované, než RVP pro gymnázia, nicméně to nebrání jeho pochopení a výkladu. V tabulce 1 jsem vyňal z RVP tabulku, týkající se tématu mé diplomové práce.

Výsledky vzdělávání	Učivo
Žák: - zná vlastnosti algoritmu; - zanalyzuje úlohu a algoritmizuje ji; - zapíše algoritmus vhodným způsobem;	1 Algoritmizace - význam, prvky algoritmu
- použije základní datové typy; - použije řídicí struktury programu; - vytvoří jednoduché strukturované programy;	2 Strukturované programování - datové typy - řídicí struktury
- rozumí pojmům třída, objekt a zná jejich základní vlastnosti; - použije jednoduché objekty;	3 Úvod do objektového programování - třída, objekt, vlastnosti tříd
- zná výhody použití jazyka SQL; - použije základní příkazy jazyka SQL;	4 Základy jazyka SQL - základní příkazy (SELECT, UPDATE, INSERT, DELETE)
- aplikuje zásady tvorby WWW stránek; - orientuje se ve struktuře HTML stránky; - vytvoří webové stránky včetně optimalizace a validace; - použije formuláře a skriptovací jazyk.	5 Tvorba statických a dynamických webových stránek

Tabulka 1 – Náplň vyučování na oboru Informační technologie, zdroj: [3]

Z tabulky 1 vyplývá, že se vyučování informatiky v rámci studijního oboru Informační technologie zaměřuje a opírá v prvních dvou bodech o výuku algoritmizace a programování.

1.2 Informatické myšlení

V připravovaných revidovaných RVP se hojně vyskytuje pojem informatické myšlení, který dostatečně vysvětluje oficiální materiál RVP níže:

„Co je informatické myšlení? Je to způsob myšlení, který se zaměřuje na popis problému, jeho analýzu a hledání efektivních řešení. Nabízí nám sadu nástrojů a postupů. Když se s nimi seznámíme a naučíme se je používat, budeme je moci uplatňovat opakovaně a v různých situacích. Informatické myšlení se učíme uplatňovat nejprve na jednoduchých, a postupně stále složitějších úlohách, problémech. Čím komplexnější problém, tím užitečnější je využití nástrojů a přístupů informatického myšlení.“ [4]

Informatické myšlení je přesný ideální nástroj, který je potřeba do budoucna u žáků rozvíjet. A skvělým nástrojem pro rozvoj informatického myšlení, je dle mého názoru, právě výuka algoritmizace a programování, jelikož dle výše uvedené citace, naplňuje důležité parametry informatického myšlení.

2 ANALÝZA VYUŽÍVANÉHO SOFTWARE PRO VÝUKU PROGRAMOVÁNÍ NA STŘEDNÍCH ŠKOLÁCH

V této kapitole se budu zabývat výzkumem, který programovací jazyk je na středních školách tím nejpoužívanějším pro výuku základů algoritmizace a programování.

2.1 Provádění výzkumu

V rámci zjišťování jsem za pomoci emailu oslovil několik středních škol a gymnázií. Taktéž jsem využil Facebookovou skupinu: „Učíme informatiku“.

Dotaz byl pokládán jednoduše:

Dobrý den,

jmenuji se Martin Majer a jsem studentem 5. ročníku oboru Učitelství informatiky pro střední školy na Fakultě aplikované informatiky Univerzity Tomáše Bati ve Zlíně.

Pro účel své diplomové práce bych potřeboval zjistit, jaký programovací jazyk se u Vás na škole používá k výuce základů algoritmizace/programování, ale jen v případě, že tyto základy vyučujete.

Jen pro úplnost myslím tím plnohodnotný programovací jazyk, jako například Pascal, C nebo Python, ne nějaké jazyky určené pro výuku programování u dětí, jako například Scratch atd.

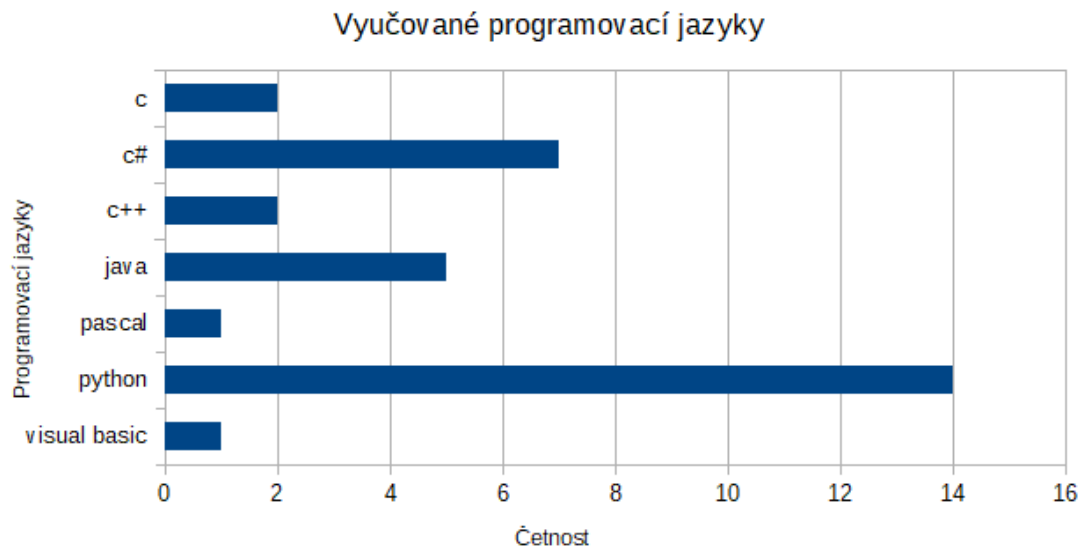
Děkuji za Váš čas,

Bc. Martin Majer

Popřípadě byl dotaz podle použité platformy dotaz lehce upravován.

2.2 Vyhodnocení výzkumu

Ze všech dostupných odpovědí na můj dotaz jsem sestavil následující graf, který je zobrazen na obrázku 1.



Obrázek 1 – Přehled vyučovaných programovacích jazyků, zdroj: vlastní

Celkově jsem dostal 32 odpovědí. Některé školy vyučují více programovacích jazyků zároveň. Z grafu uvedeném na obrázku 1 je na první pohled patrné, že nejpoužívanější jazyk pro výuku algoritmizace a programování na středních školách je jazyk Python.

Dále se tedy v mé práci budu zaměřovat, na jazyk Python.

3 POPIS POUŽÍVÁNÉHO SOFTWARE

Jako nejpoužívanější programovací jazyk pro výuku programování na středních školách a gymnáziích se podle výzkumu jeví jazyk Python. Toto zjištění není překvapující. Python se v poslední době těší velké oblibě, jednak mezi laickou veřejností ale i mezi profesionály, kteří se programováním žijí.

Nyní si nastíníme, co se vlastně může skrývat pod pojmy programování a algoritmizaci resp. algoritmem, abychom poté mohli jednotně tvořit další výukové materiály.

„Umět programovat neznamena prostě vytvářet programy pro počítač – znamená vytvářet co nejefektivnější, co nejrychlejší a nejjednodušší a přitom dokonale funkční programy. Je určitě možné naučit se komunikovat s počítačem tak, aby nám rozuměl a aby plnil naše rozkazy.“ [5]

„Pojem algoritmus vznikl v matematice a používal se až do nedávna k řešení úloh matematických a logických. Velkým úspěchem vědy je, že ukázala algoritmickou podstatu mnohých nematematických úloh.“ [6]

3.1 Python

Co je tedy Python?

Python je moderní univerzální programovací jazyk, který se vyznačuje zejména jednoduchou syntaxí, díky které se snadno učí a posléze používá. Jeho univerzálnost ho předurčuje k používání v rozličných sférách, ať už při tvorbě konzolových či okenních aplikací, tvorbě webů či programování jednoduchých mikropočítačů jako například Arduino. Tento výčet je samozřejmě v praxi mnohem rozsáhlejší a komplexnější. Další důležitou vlastností jazyku Python je to, že se jedná o tzv. interpretovaný jazyk. *„Tyto jazyky pro překlad programu používají tzv. interpreter. Zdrojový (textový) soubor je čten a překládán řádek po řádku. Nevýhodou je nízká rychlost, což při dnešním výkonu počítačů nehraje roli.“ [7]* Toto je jeden z hlavních rozdílů oproti jiným, v praxi běžně používaným jazykům, jako například jazyk C.

Podle Vitáska (2012, zdroj: [8]), je pro úvodní kurzy, zabývající se programováním, tedy pro úplné programátorské začátečníky, typické to, že se zabývají programováním jako tako-

vým a ne konkrétním programovacím jazykem. Ten je nutné naučit se ještě nad rámec samotného programování, tedy začátečnickům to komplikuje vstup do programátorského světa. Programovací jazyk je pouze prostředkem pro vyjádření jejich myšlenek.

Z této myšlenky plyne, že je vhodné učit takový programovací jazyk, kterého učení příliš nekomplikuje samotné učení programování. Tedy hodí se jazyk s jednoduchým zápisem a přehlednou strukturou, což Python bez pochyby je.



Obrázek 2 – Logo jazyku Python, zdroj: [14]

3.1.1 Historie jazyku Python

Podle informací dostupných na webových stránkách jazyku Python (zdroj: [9]), tento programovací jazyk vzniknul na počátku 90. let 20. století. Za jeho otce je považován Guido van Rossum, který v té době pracoval v Matematickém centru Stichting v Nizozemsku. Tento programovací jazyk je považován za následníka programovacího jazyka ABC. Guido van Rossum je i v dnešní době stále hlavním autorem nejnovějších verzí jazyka Python, avšak má velké množství spolupracovníků. V roce 2001 byla vytvořena nezisková společnost s názvem Python Software Foundation, která disponuje duševním vlastnictvím spojeným s programovacím jazykem Python. Všechna vydání jazyku Python jsou vydávány v rámci licence Open source. „*Pojem open source (v češtině někdy jako otevřený software) se používá pro označení programu nebo systému s volně přístupným zdrojovým kódem, který tak může každý libovolně upravovat a měnit za účelem vývoje daného nástroje.*“ [10] Aktuální verze programovacího jazyku, používanou i v této diplomové práci, je verze 3.9.2.

3.1.2 Výhody a nevýhody jazyku Python

Na základě informací dostupných na výukových webových stránkách Scholarlyoa (zdroj: [11]), jsem vybral hlavní výhody a hlavní nevýhody používání programovacího jazyku Python.

3.1.2.1 Výhody

- Jednoduchá a přehledná syntaxe
- Dynamický styl zápisu
- Není třeba využívat pro psaní kódu tolika závorek (na rozdíl od jazyku C)
- Automatické přidělování paměti
- Aktivní komunita
- Velké množství knihoven

3.1.2.2 Nevýhody

- Jelikož je Python jazykem interpretovaným, může být oproti jiným jazykům pomalejší a tudíž méně vhodný pro větší a složitější aplikace právě z důvodu rychlosti
- Python je vyšším programovacím jazykem, takže není vhodný psaní programů na úrovni hardwaru
- U některých speciálních úkolů, může být automatické přidělování paměti nevýhodou

3.2 IDLE

Toto konkrétní IDE (prostředí), bylo vybráno z důvodu toho, že je dostupné po instalaci Pythonu na operačním systému Microsoft Windows. Prostředí je jednoduché a obsahuje všechny potřebné položky, jako například debugger atd.

Dle informací dostupných na webových stránkách jazyku Python (zdroj: [12]), je IDLE integrované výukové a vývojové prostředí určené pro práci s programovacím jazykem Python. Mezi jeho nejdůležitější vlastnosti patří možnost programování v čistém Pythonu za pomoci nástrojů thinter GUI toolkit. Vývojové prostředí IDLE je taktéž multiplatformní, funguje až na výjimky stejně v operačních systémech Windows, Unix i macOS. Dále bych zmínil širokou přizpůsobitelnost vývojového prostředí, možnost pracovat na více oknech programu zároveň. V každém okně je možnost vícenásobného vrácení zpět.

3.2.1 Výhody a nevýhody prostředí IDLE

Na základě informací z webové stránky TrustRadius (zdroj: [13]), která shromažďuje recenze uživatelů na různé počítačové programy, jsem vybral nejzásadnější výhody a nevýhody prostředí IDLE od lidí přímo z praxe. Jen bych ještě podotknul, že podle uvedeného webu má IDLE hodnocení 8,4/10, což je velmi slušné.

3.2.1.1 Výhody

- Uživatelsky přívětivé prostředí
- Dobrý vestavěný debugger
- Doplnování klíčových slov
- Dobře přizpůsobitelné prostředí

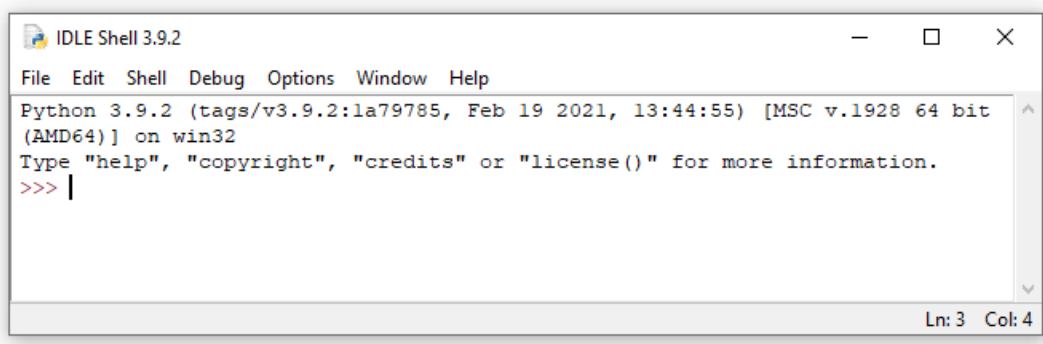
3.2.1.2 Nevýhody

- Možná až moc minimalistické grafické rozhraní
- Dostupný je jenom základní debugger s omezenými funkcemi
- Omezená škála výstupů

3.2.2 Práce v IDLE

Podrobný návod pro práci s prostředím IDLE naleznete v příloze P3 na CD s názvem `idle_navod.docx`.

V tomto materiálu je uveden návod, kde přes stažení a instalaci samotného vývojového prostředí, se čtenář dostane k podrobnému popisu samotného programu a poté i tvorbě samostatných jednoduchých, ale i složitějších programů. Vše je doprovázeno obrázky z příslušného programu.



Obrázek 3 – IDLE Shell 3.9.2, zdroj: vlastní

4 TVORBA VÝUKOVÝCH MATERIÁLŮ

Celý vyučovací blok se dá rozdělit na dvě části, tedy na algoritmizaci a programování. Podle časového harmonogramu uvedeného v Tabulce 2 na straně 24 je zřejmé, že větší prostor dostala algoritmizace. Tedy ona je hlavní položkou tohoto bloku. Programování disponuje menší časovou dotací a v něm se jedná hlavně o pochopení základů, které budou podrobněji rozváděny později, což již není tématem tohoto vyučovacího bloku. Žáci se v tomto bloku naučí podrobně algoritmizaci a poté základy programování.

Celkově výukové materiály obsahují rozdělení vyučovacích hodin, podpůrné prezentace, testy a návody pro práci s příslušnými programy.

Nejprve jsou v materiálu popsány obecné informace jako cíle a formy výuky, vyučovací metody, klíčové kompetence atd. Poté následuje přehledné rozdělení jednotlivých hodin.

Každá hodina začíná tématem vyučovací hodiny, poté je uveden cíl vyučovací hodiny a posléze i organizační forma vyučovací hodiny. Následuje tabulka, kde je průběh hodiny v časovém rámci podrobně popsán. Nechybí ani uvedení tématu odborného obsahu a použité pomůcky.

Následně jsou tyto informace slovně rozvedeny, dále obsahuje materiál pokyny pro vyučujícího, kde je instruován, jaké materiály má při výuce používat. Jsou zde uvedeny úkoly pro žáky, které jsou i vyřešeny.

4.1 Algoritmizace

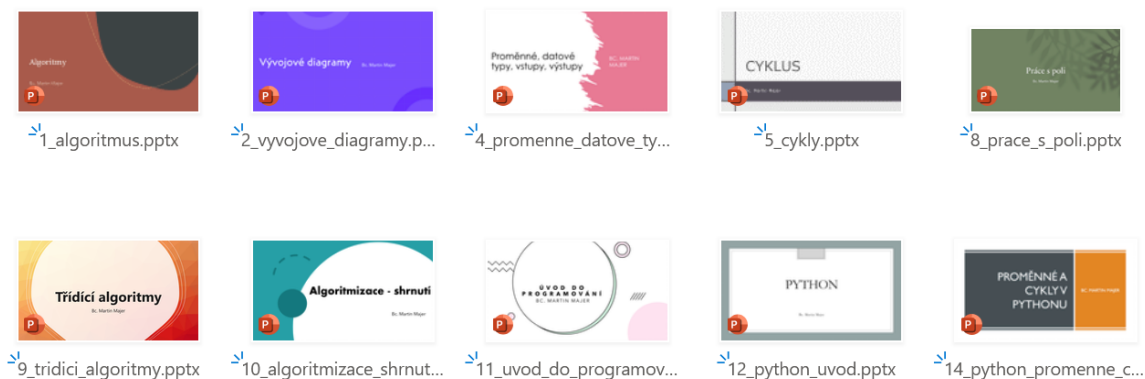
V části algoritmizace nejprve začínáme úvodní hodinou, kde se žáci dozvědí, co si pod pojmem algoritmizace představí. Dále žáci budou probírat vývojové diagramy. „*Vývojový diagram je symbolický algoritmický jazyk, který se používá pro názorné zobrazení algoritmu a je to jedna z nejdokonalejších forem zápisu algoritmu.*“ [15] Poté se znalostmi vývojových diagramů již budou moci v následující hodině samostatně tvořit algoritmy zapsané pomocí vývojových diagramů. Postupně se žáci ponoří do problematiky používání proměnných, datových typů vstupů a výstupů. Posléze následují 3 hodiny zaměřené na tvorbu cyklů. Se znalostí tvorby cyklů budou žáci schopni tvořit v následující hodině i složitější algoritmy prakticky. Poté se ještě dozvědí něco o třídících algoritmech a zaměří se konkrétně na Bubble sort. V předposlední hodině této podkapitoly se shrne všechno dosavadní učivo probrané v rámci podkapitoly. V poslední hodině se ověří nabyté znalosti testem.

4.2 Programování

V části programování začneme nejprve tím, co si pod pojmem programování lze představit. Poté se již konkrétně přesuneme na úvodní hodinu, kde se žáci seznámí s jazykem Python. Po tomto úvodu již následuje první programování v novém programovacím jazyku a prostředí. Po prvních krůčcích si žáci začnou zkoušet zápisy proměnných a jednotlivých cyklů. Následně si žáci vyzkouší tvorbu jednoduchých programů. Posléze zavzpomínají na tvorbu vývojových diagramů, jelikož se budou učit přepisovat vývojové diagramy do programovacího jazyka. Poté přicházejí 2 hodiny, kde žáci budou tvořit komplexnější programy. Žáci budou schopni používat i knihovny. „*Termínem knihovna se označuje soubor s podprogramy či třídami využitelnými v celé řadě dalších programů. Jestliže chce program některý z podprogramů či tříd knihovny využívat, musí knihovnu připojit* „ [16] Na závěr přichází test nabytých znalostí v 2. podkapitole.

4.3 Tvorba samotných výukových materiálů

Značnou část vytváření práce zabrala tvorba prezentací. Bylo vytvořeno, jak je to patrné z obrázku 4, celkem 10 výukových podpůrných prezentací, z nichž 7 se zabývá částí algoritmizace a zbylé 3 části programování.



Obrázek 4 – Výukové prezentace, zdroj: vlastní

Každá prezentace má záměrně vybrané jiné pozadí, protože zastávám názor, že stejné pozadí všech prezentací by na žáky ve výuce působilo příliš fádně a nudně. Použití rozličných pozadí ve výukových prezentacích bude na pozornost žáků působit pozitivně, jednotlivé prezentace nesplynou v celek a zmatek. Odlišné pozadí pomůže rozřadit získané vědomosti z každé probrané oblasti.

II. PRAKTICKÁ ČÁST

5 PODKLADY PRO VÝUKU

V kapitole 5 uvádím kompletní rozvržení hodin pro výuku algoritmizace a programování. Materiál začíná na straně 22 a pokračuje až na stranu 77. Obsah tohoto konkrétního materiálu je podrobně rozebrán v kapitole 4.

Celkově bych ještě uvedl seznam všech příloh, které jsem pro výuku vytvořil. Všechny tyto přílohy jsou přiloženy jako příloha P3 na CD.

Jedná se tedy o:

- Návod pro práci v prostředí IDLE
- Rozvržení vyučovacích hodin (uvedeno kompletně v této kapitole od strany 22 až po stranu 77)
- Test na shrnutí části algoritmizace ve dvou variantách
 - Verze bez odpovědí
 - Verze se správnými odpověďmi (příloha P1)
- Test na shrnutí části programování ve dvou variantách
 - Verze bez odpovědí
 - Verze se správnými odpověďmi (příloha P2)
- Výukové prezentace
 - Algoritmus
 - Vývojové diagramy
 - Proměnné, datové typy
 - Cykly
 - Práce s poli
 - Třídící algoritmy
 - Algoritmizace – shrnutí
 - Úvod do programování
 - Python – Úvod
 - Python – Proměnné, cykly

ŘEŠENÉ ÚLOHY Z OBLASTI ZPRACOVÁNÍ INFORMACÍ: ALGORITMIZACE ÚLOH

Vytvořil: Bc. Martin Majer

Předmět: Informační technologie (ICT, IVT, IT)

Téma: algoritmizace, programování

Cílová skupina: žáci střední školy/gymnázia

Cíle:

- Kognitivní – Žák se naučí základní problematiku algoritmizace a programování. Žák je schopen vlastními slovy tuto problematiku popsat a shrnout. Žák využije tyto znalosti buď v dalším studiu, nebo v řešení reálných situací. Žák je schopen o tématu diskutovat.
- Afektivní - Žák během výkladu pozorně naslouchá probírané látce, aby ji později byl schopen aplikovat ve svém osobním, studijním či profesním životě. Žák si uvědomuje, jaké postupy řešení daných problémů on zná a vybírá, jaké by použil. Žák je schopen v této problematice radit jiným.
- Psychomotorické – Žák zlepšuje svou schopnost ovládat počítač.

Formy: frontální výuka, samostatná práce

Metody: slovní metody (vysvětlování, přednáška), názorně demonstrační metody (předvádění, pozorování, instruktáž), metody dovednostní praktické (napodobování, vytváření)

Pomůcky: počítače, projektor, interaktivní tabule

Lokace: počítačová učebna

Časová dotace: 19 vyučovacích hodin

Klíčové kompetence

1. K učení – Žák se naučí, jak se tvoří algoritmy. Žák se naučí jak napsat jednoduchý program.
2. Komunikativní - Žák bude schopen předat nabitě znalosti v oblasti algoritmizace a programování i jiným lidem.
3. Řešení problémů - Žák bude schopen samostatně odpovědět na zadaný úkol.
4. Občanská - Žák si bude vědom výhod a nevýhod jiných programovacích jazyků používaných jinými lidmi.
5. Sociální - Žák dokáže poradit ohledně tvorby jednoduchých programů i lidem nezkušeným v prostředí informatiky.
6. Pracovní - Žák bude schopen vyřešit daný úkol.

Časový harmonogram

Číslo hodiny	Téma hodiny
1.	Algoritmus - úvodní hodina
2.	Vývojové diagramy
3.	Algoritmus - zápis
4.	Proměnné, datové typy, vstupy, výstupy
5.	Cykly I.
6.	Cykly II.
7.	Cykly III.
8.	Procvičování tvorby složitějších algoritmů
9.	Třídící algoritmy – Bubble sort
10.	Algoritmizace - shrnutí
11.	Test - algoritmizace, Úvod do programování
12.	Python - úvod
13.	Python – první programování
14.	Python – zápis proměnných, cyklů
15.	Python – tvorba jednoduchých programů
16.	Python – přepisování vývojových diagramů
17.	Python – tvorba komplexního programů I.
18.	Python – tvorba komplexního programů II.
19.	Test - Python

Tabulka 2 – Časový harmonogram vyučovacích hodin

1.

Téma vyučovací hodiny:**Algoritmus - úvodní hodina****Cíle vyučovací hodiny:****Definovat, popsat, vysvětlit****Organizační forma vyučovací hodiny:****Frontální výuka**

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci	Algoritmy	Počítače Projektor Interaktivní tabule
5	Administrace			
10	Prezentace na téma algoritmus	Naslouchání prezentaci a tvorba poznámek		
15				
20				
25				
30				
35	Odpovídání na dotazy	Kladení dotazů a nejasností		
40	Shrnutí			
45				

Tabulka 3 – Vyučovací hodina: Algoritmus - úvodní hodina

Popis vyučovací hodiny:

Tématem této vyučovací hodiny je: algoritmus – úvodní hodina. To značí, že jde o úvodní hodinu z prvního cyklu, tedy z algoritmizace.

Odborným obsahem je v této hodině oblast algoritmizace.

Žáci se tuto hodinu naučí definovat pojem algoritmus, popsat jeho vlastnosti a vysvětlit jeho funkci.

Celá vyučovací hodiny proběhne ve formě frontální výuky s možností dotazů od žáků.

Pokyny pro vyučujícího:

Vyučující využívá pro výuku prezentaci s názvem: 1_algoritmus.pptx

Brainstorming - algoritmus

2.

Téma vyučovací hodiny:**Vývojové diagramy****Cíle vyučovací hodiny:****Interpretovat, řešit****Organizační forma vyučovací hodiny:****Frontální výuka + samostatná práce**

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci	Vývojové diagramy	Počítače Projektor Interaktivní tabule
5	Administrace			
10	Zopakování vlastností algoritmů	Naslouchání prezentaci a tvorba poznámek		
15				
20	Prezentace na téma vývojových diagramů	Naslouchání prezentaci a činění poznámek		
25				
30	Společné používání vývojových diagramů	Společné používání vývojových diagramů		
35				
40	Odpovídání na dotazy	Kladení dotazů a nejasností		
45				

Tabulka 4 – Vyučovací hodina: Vývojové diagramy

Popis vyučovací hodiny:

Tématem této vyučovací hodiny jsou vývojové diagramy, zejména to jak se algoritmy zapisují a jak se dále využívají.

Odborným obsahem je v této hodině oblast vývojových diagramů.

Žáci se tuto hodinu naučí interpretovat tvorbu vývojového diagramu a budou schopni řešit zadané úkoly na téma tvorby vývojových diagramů.

Vyučovací hodina proběhne ve formě frontální výuky, kdy bude prezentována prezentace a poté bude společně vyučující s žáky řešit jednoduché formy zápisu vývojových diagramů, kde bude figurovat při osvojování nových dovedností a znalostí i samostatná práce.

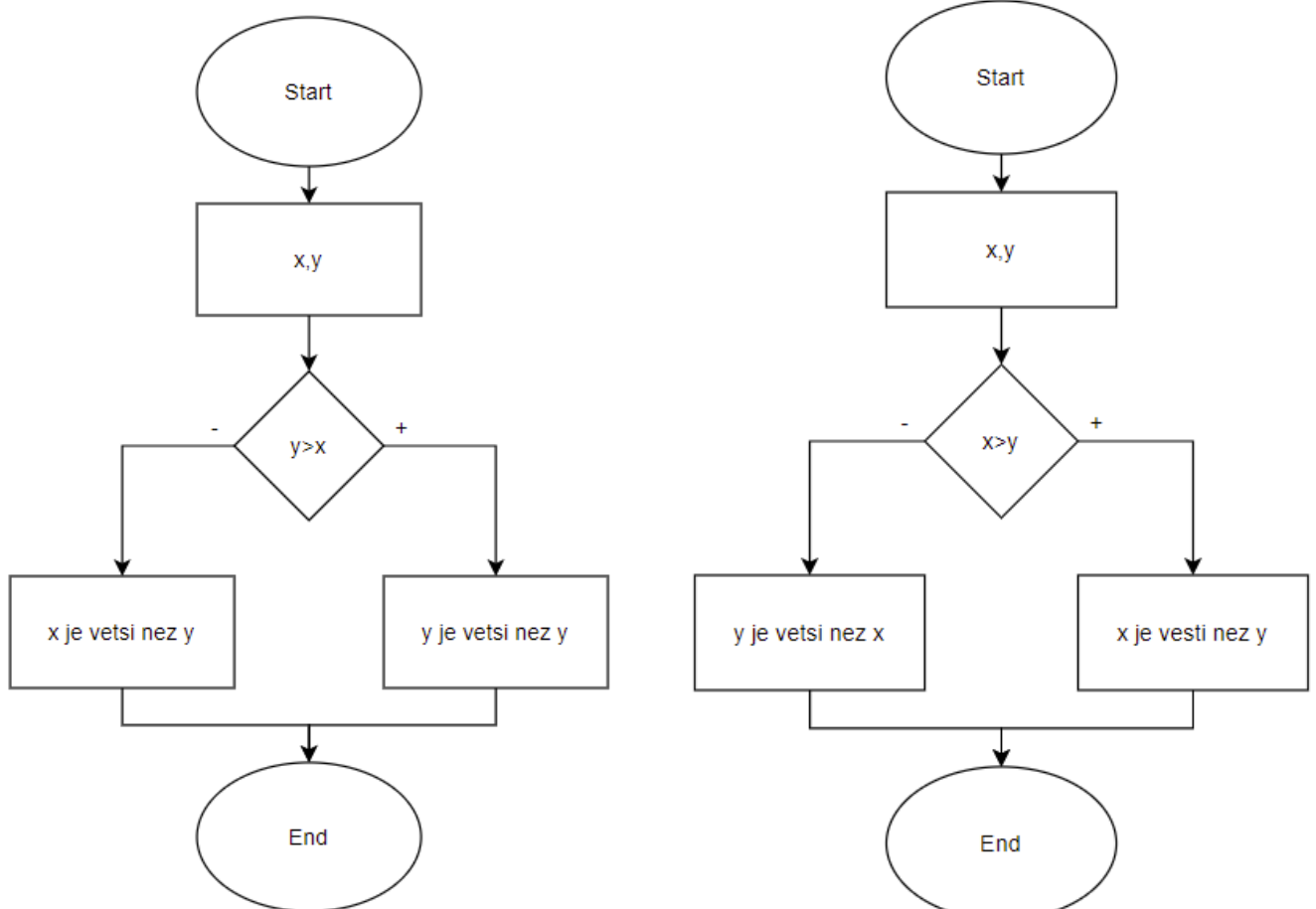
Pokyny pro vyučujícího:

Vyučující využívá pro výuku prezentaci s názvem: 2_vyvojove_diagramy.pptx

Úkol: Vytvořte algoritmus, zapsaný pomocí vývojového diagramu, který vypíše z dvojice čísel x a y , které číslo je větší než to druhé.

Řešení:

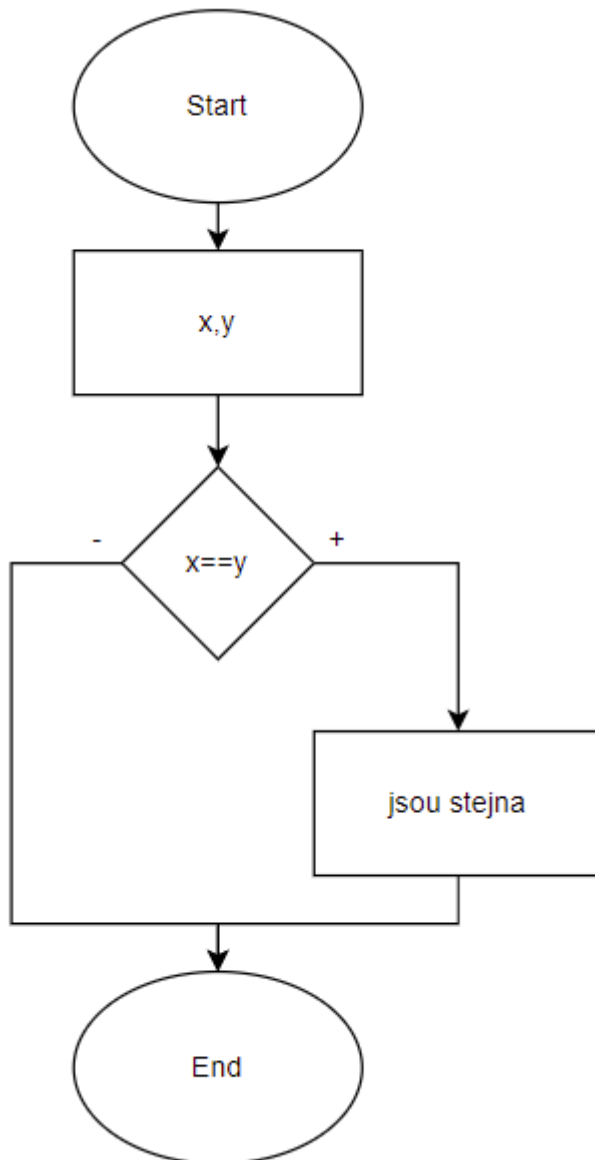
obě řešení jsou platná



Úkol: Vytvořte algoritmus, zapsaný pomocí vývojového diagramu, který vypíše hlášku „jsou stejná“, pokud obě čísla z dvojice x a y si budou rovna. Pokud budou od sebe navzájem různá, nevypíše se nic.

Řešení:

Rovnost se zapisuje $==$, nerovnost se zapisuje $!=$



3.

Téma vyučovací hodiny:**Algoritmus - zápis****Cíle vyučovací hodiny:****Charakterizovat, vyzkoušet****Organizační forma vyučovací hodiny:*****Samostatná práce***

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci	Algoritmy	Počítače Projektor Interaktivní tabule
5	Administrace			
10	Zadání samostatné práce	Samostatná tvorba jednotlivých vývojových diagramů		
15	Společné nastínění řešení.			
20				
25				
30	Kontrola výsledků práce			
35				
40	Dotazy	Kladení dotazů a nejasností		
45	Shrnutí			

Tabulka 5 – Vyučovací hodina: Algoritmus - zápis

Popis vyučovací hodiny:

Tématem této vyučovací hodiny je zápis algoritmů.

Odborným obsahem je v této hodině oblast algoritmizace.

Žáci shrnou své znalosti v zadané samostatné práci, kde si vyzkouší znalosti nabyté v předchozích vyučovacích hodinách.

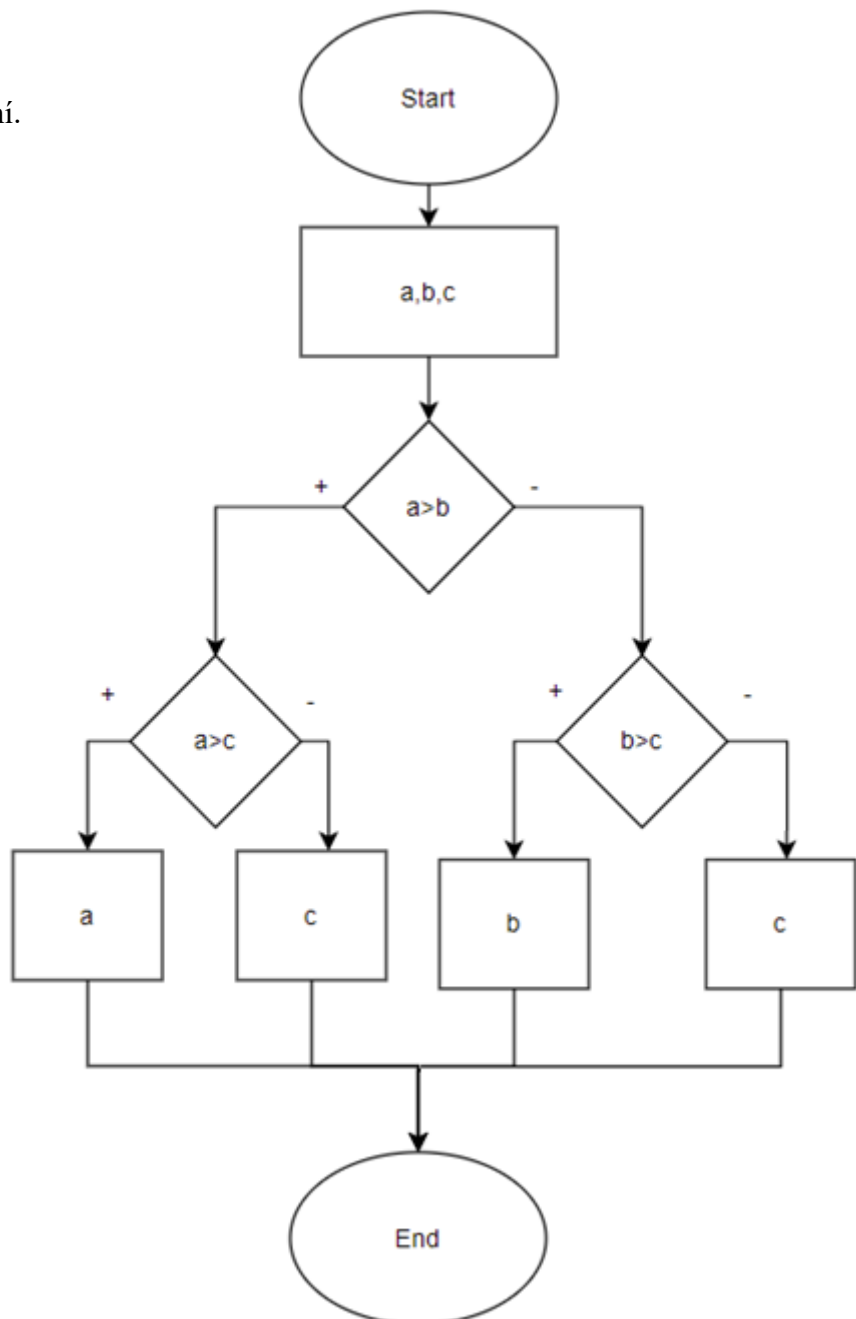
Celá vyučovací hodina proběhne až na výjimku jako samostatná práce, kde na konci bude tato práce zkontrolována.

Pokyny pro vyučujícího:

Úkol: Vytvořte algoritmus zapsaný pomocí vývojového diagramu, který vypíše nejvyšší číslo z trojice a , b a c .

Řešení:

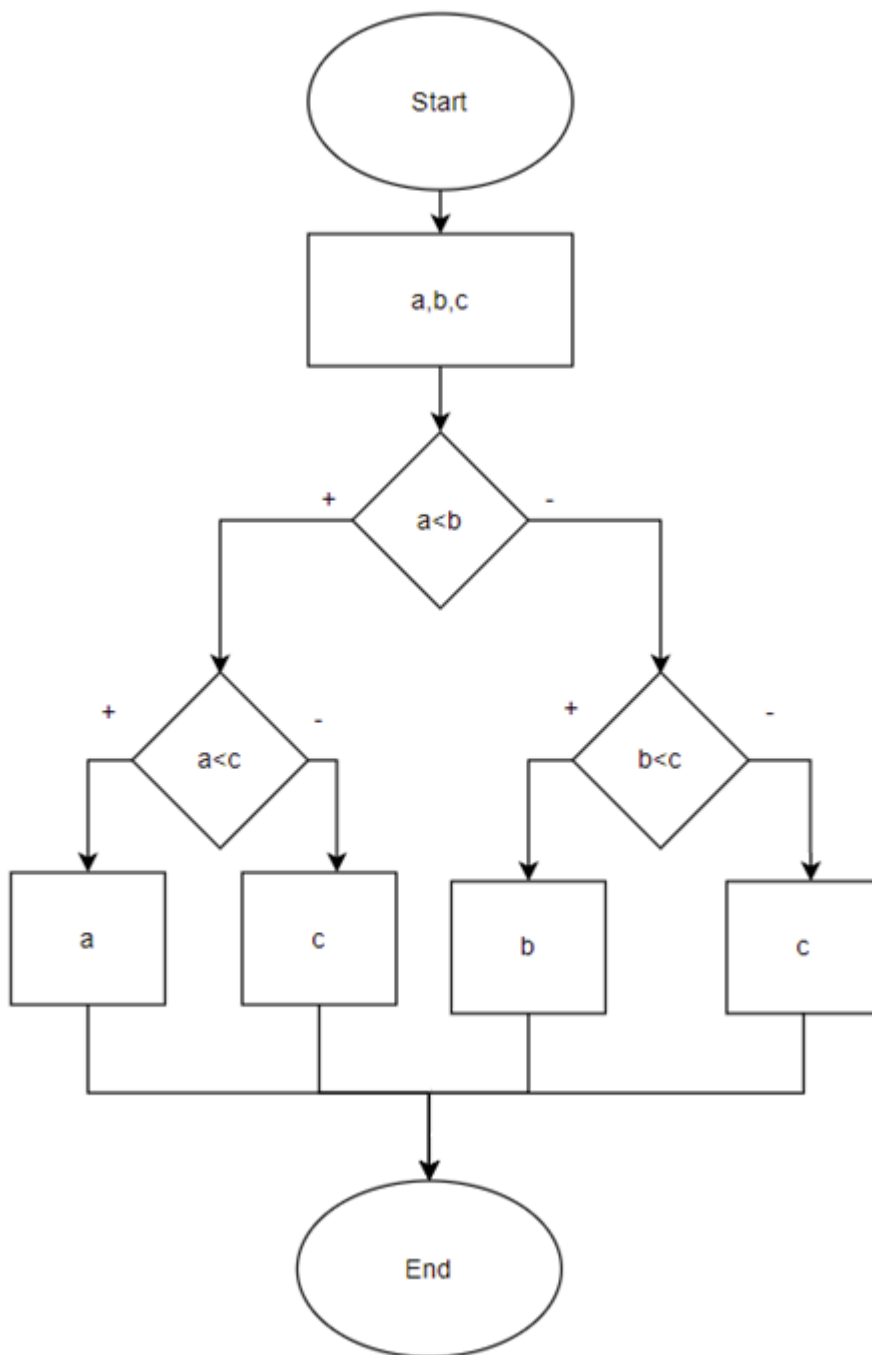
Společně nastítnit řešení.



Úkol: Vytvořte algoritmus zapsaný pomocí vývojového diagramu, který vypíše nejnižší číslo z trojice a, b a c.

Řešení:

stačí přepsat všechny 3 znaménka „>“ na „<“



4.

Téma vyučovací hodiny: Proměnné, datové typy, vstupy, výstupy

Cíle vyučovací hodiny: Vyjádřit, říci vlastními slovy

Organizační forma vyučovací hodiny: *Frontální výuka + samostatná práce*

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci		
5	Administrace			Počítače Projektor Interaktivní tabule
10	Prezentace na téma proměnné, datové typy, vstupy, výstupy	Naslouchání prezentaci a tvorba poznámek	Algoritmizace - proměnné, datové typy, vstupy, výstupy	
15				
20				
25				
30	Zadání samostatné práce	Vypracovávání samostatné práce		
35				
40	Kontrola výsledků samostatné práce			
45				

Tabulka 6 – Vyučovací hodina: Proměnné, datové typy, vstupy, výstupy

Popis vyučovací hodiny:

Tématem této vyučovací hodiny jsou proměnné, datové typy, vstupy, výstupy.

Odborným obsahem je v této hodině oblast algoritmizace.

Žáci se tuto hodinu naučí vyjádřit své algoritmy korektně za použití správných datových typů. Dále se naučí používat proměnné a tyto proměnné načítat a vypisovat.

Vyučovací hodina začne ve formě frontální výuky s možnostmi dotazů od žáků. Ve zbytku času následuje samostatná práce – procvičování.

Pokyny pro vyučujícího:

Vyučující využívá pro výuku prezentaci s názvem: 4_promenne_datove_typy.pptx

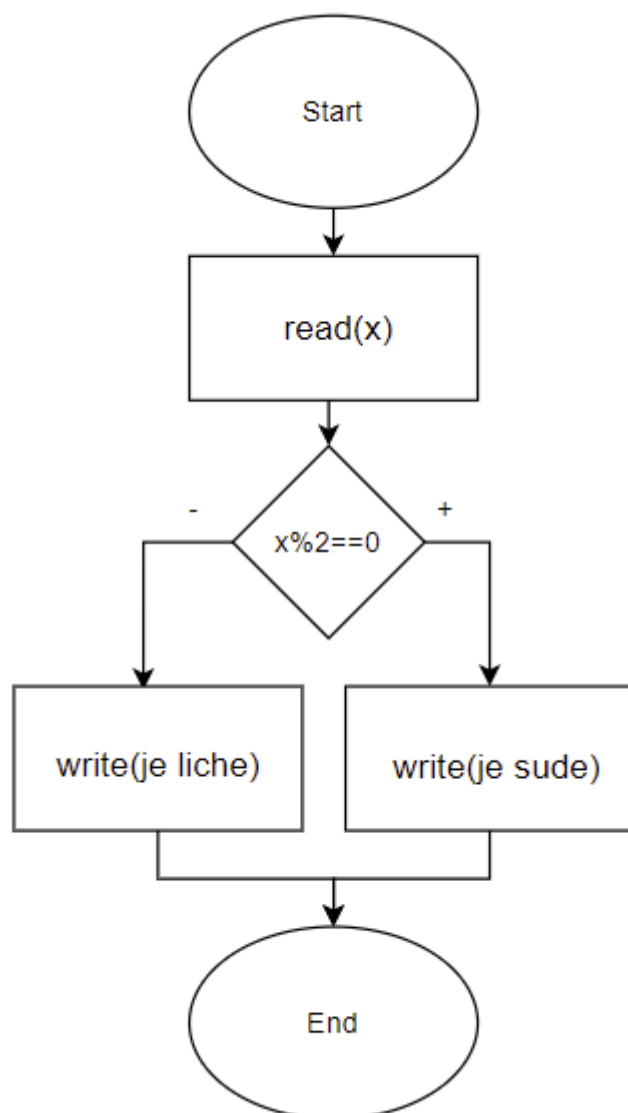
Procvičování:

$10\%2 = 0$
 $10\%8 = 2$
 $66\%32 = 2$
 $66\%35 = 31$
 $100\%25 = 22$
 $100\%11 = 1$

Jak zjistíme, jestli je číslo sudé?
=> Je dělitelné dvěma beze zbytku => $\text{číslo}\%2 = 0$

Úkol: Vytvořte algoritmus zapsaný pomocí vývojového diagramu, který zjistí, zda je načítané číslo x sudé nebo liché a tuto vlastnost vypíše.

Řešení:
Využití modula



5.

Téma vyučovací hodiny:**Cykly I.****Cíle vyučovací hodiny:****Definovat, popsat****Organizační forma vyučovací hodiny:****Frontální výuka**

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci	Cykly	Počítače Projektor Interaktivní tabule
5	Administrace			
10	Prezentace na téma cyklů	Naslouchání prezentaci a tvorba poznámek		
15				
20				
25				
30				
35				
40	Dotazy	Kladení dotazů a nejasností		
45				

Tabulka 7 – Vyučovací hodina: Cykly I.

Popis vyučovací hodiny:

Tématem této vyučovací hodiny jsou cykly, které jsou celkem rozděleny na 3 hodiny, takže toto je úvod do tematiky cyklů.

Odborným obsahem v této hodině jsou cykly.

Žáci se tuto hodinu naučí definovat pojem cyklus a popsat jednotlivé druhy cyklů.

Celá vyučovací hodina proběhne ve formě frontální výuky s možnostmi dotazů od žáků.

Pokyny pro vyučujícího:

Vyučující využívá pro výuku prezentaci s názvem: 5_cykly.pptx

6.

Téma vyučovací hodiny:**Cykly II.****Cíle vyučovací hodiny:****Definovat, popsat, aplikovat****Organizační forma vyučovací hodiny:****Frontální výuka + samostatná práce**

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci	Cykly	Počítače Projektor Interaktivní tabule
5	Administrace			
10	Zopakování cyklů	Naslouchání prezentaci a tvorba poznámek		
15	Společná tvorba jednotlivých cyklů	Společná tvorba jednotlivých cyklů		
20				
25				
30				
35				
40	Dotazy	Kladení dotazů a nejasností		
45				

Tabulka 8 – Vyučovací hodina: Cykly II.

Popis vyučovací hodiny:

Tématem této vyučovací hodiny jsou opět cykly.

Odborným obsahem v této hodině jsou cykly.

Žáci se tuto hodinu zopakují základní druhy cyklů. Po hodině už mají umět cyklus definovat, popsat jeho vlastnosti a aplikovat cykly při tvorbě vývojových diagramů.

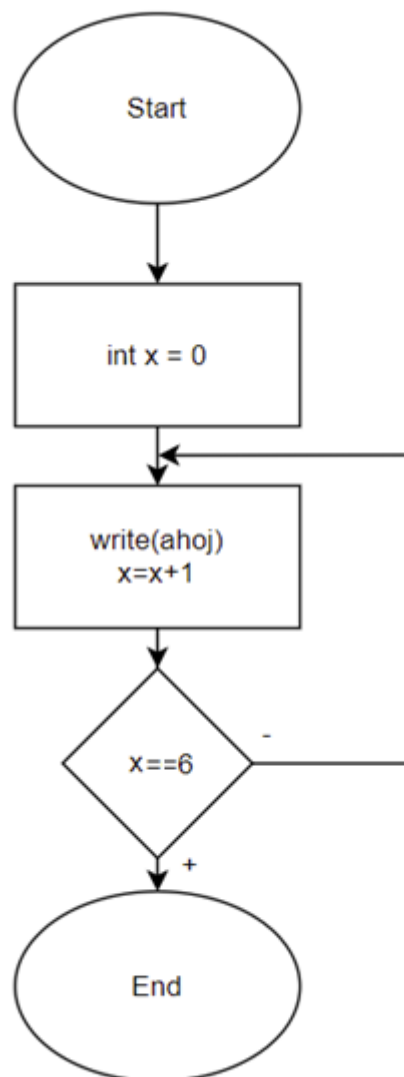
Celá vyučovací hodina proběhne ve formě frontální výuky, kde vyučující zopakuje základní vlastnosti cyklů a poté ve formě společné práce, kde bude vyučující společně se žáky tvořit jednotlivé cykly.

Pokyny pro vyučujícího:

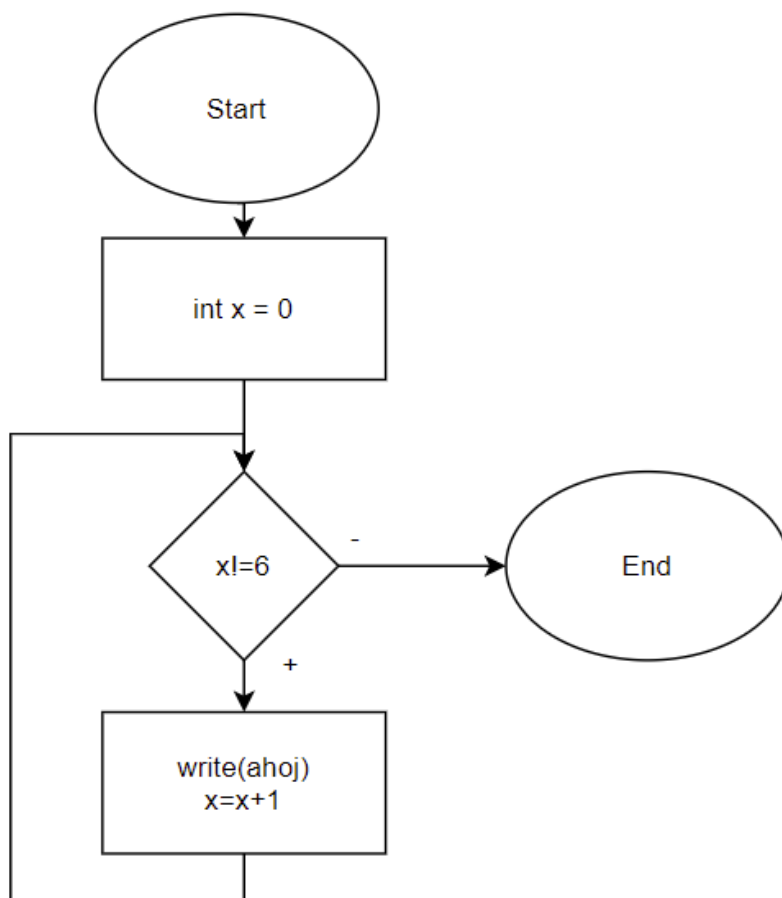
Úkol: Vytvořte algoritmus zapsaný pomocí vývojového diagramu, který 6x po sobě za pomoci cyklu vypíše ahoj. Nejprve použijte cyklus do-while, poté cyklus while a na koce cyklus for.

Řešení:

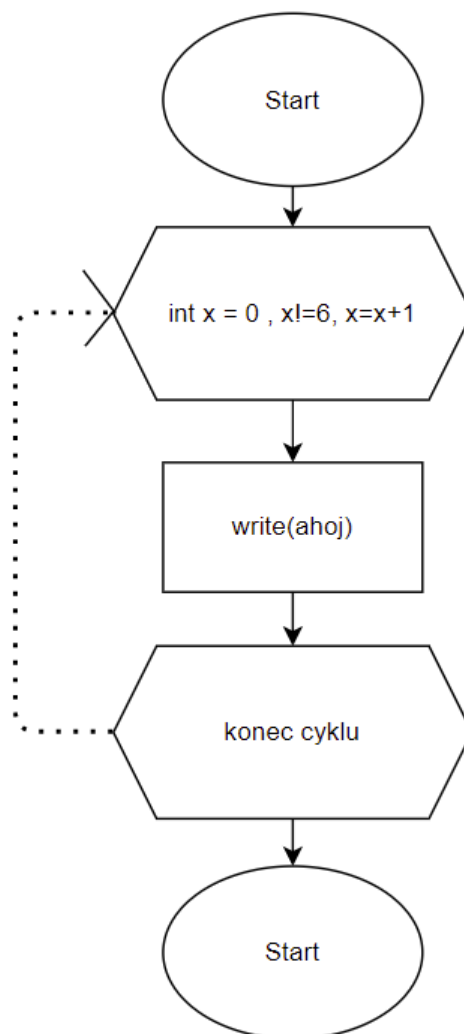
Cyklus do-while:



Cyklus while:



Cyklus for:



7.

Téma vyučovací hodiny:**Cykly III.****Cíle vyučovací hodiny:****Použít, porovnat, rozlišit****Organizační forma vyučovací hodiny:*****Samostatná práce***

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci	Cykly	Počítače Projektor Interaktivní tabule
5	Administrace			
10	Zopakování cyklů	Naslouchání prezentaci a tvorba poznámek		
15	Zadání samostatné práce	Samostatná tvorba jednotlivých cyklů		
20				
25				
30				
35	Kontrola výsledků práce	Kontrola výsledků práce		
40	Dotazy	Kladení dotazů a nejasností		
45				

Tabulka 9 – Vyučovací hodina: Cykly III.

Popis vyučovací hodiny:

Tématem této vyučovací hodiny je poslední část podkapitoly cyklů.

Odborným obsahem v této hodině jsou cykly.

Žáci budou po této hodině schopni použít jednotlivé druhy cyklů, porovnat cykly mezi sebou a rozlišit, kdy se jaký druh cyklu hodí.

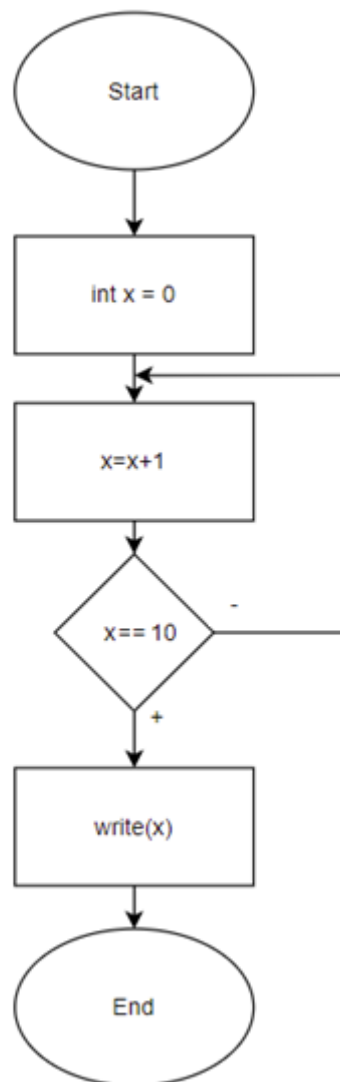
Celá vyučovací hodina proběhne ve formě samostatné práce, kdy po úvodním letmém zopakování tvorby cyklů, následuje samostatná práce, kdy budou žáci tvořit cykly samostatně a poté se společně zkontrolují.

Pokyny pro vyučujícího:

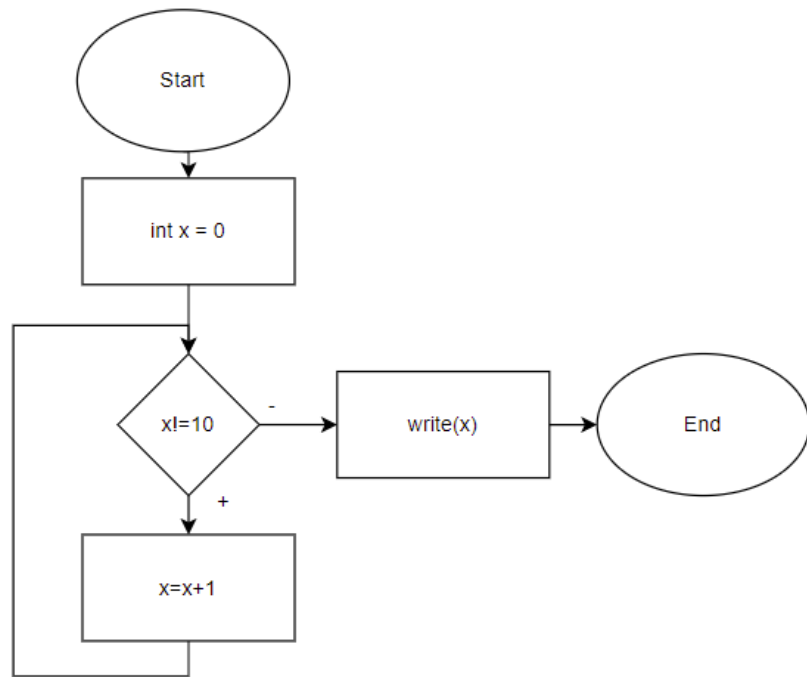
Úkol: Vytvořte algoritmus zapsaný pomocí vývojového diagramu, který na začátku bude obsahovat proměnnou $x = 0$, proměnná $10x$ projde cyklem, přičemž při každém průchodu cyklem se zvýší její hodnota o 1. Po konci cyklu ($x=10$) se hodnota x vypíše. Použijte nejprve cyklus do-while, poté while a nakonec for.

Řešení:

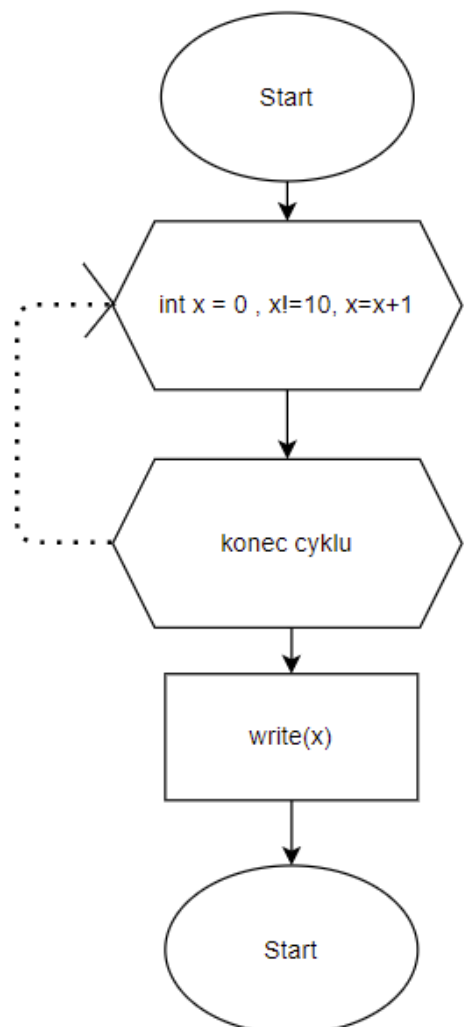
Cyklus Do-while:



Cyklus While:



Cyklus For:



8.

Téma vyučovací hodiny: Procvičování tvorby složitějších algoritmů
Cíle vyučovací hodiny: Interpretovat, vyřešit
Organizační forma vyučovací hodiny: *Frontální výuka + Samostatná práce*

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci		
5	Administrace			Počítače Projektor Interaktivní tabule
10	Prezentování prezentace ohledně práce s poli	Naslouchání prezentaci a tvorba poznámek	Tvorba složitějších algoritmů	
15	Zadání samostatné práce	Práce na samostatné práci		
20				
25				
30				
35				
40	Kontrola samostatné práce	Prezentace výsledků		
45				

Tabulka 10 – Vyučovací hodina: Procvičování tvorby složitějších algoritmů

Popis vyučovací hodiny:

Tématem této vyučovací hodiny je procvičování tvorby složitějších algoritmů.

Odborným obsahem v této hodině jsou algoritmy.

Žáci se tuto hodinu naučí interpretovat nabyté znalosti (vývojové diagramy, cykly atd.) a vyřešit za pomoci jejich znalosti složitější algoritmické úlohy.

Na začátku hodiny proběhne frontální výukou vysvětlení práce s poli. Dále již bude vyučovací hodina probíhat jako samostatná práce, kdy bude zadána samostatná práce, která bude posléze zkontrolována společně.

Pokyny pro vyučujícího:

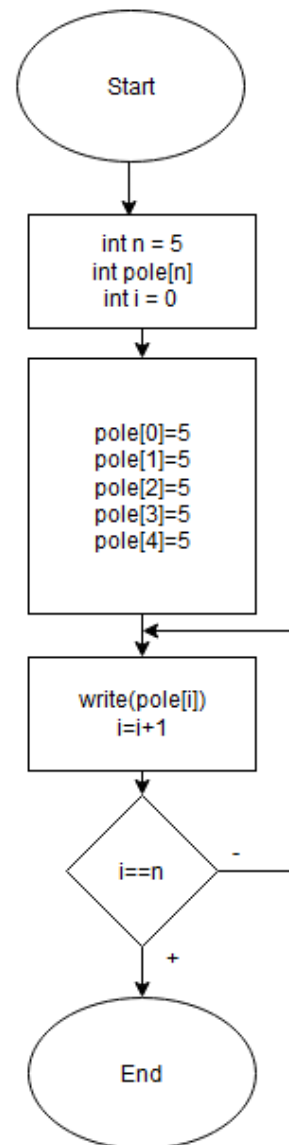
Vyučující využívá pro výuku prezentaci s názvem: 8_prace_s_poli.pptx

Úkol: Vytvořte algoritmus zapsaný pomocí vývojového diagramu, kde na začátku vytvoříte pole s 5 pozicemi. Do všech pozic ručně zapíšete číslici 5. Poté za pomoci libovolného cyklu celé pole vypíšete.

Řešení:

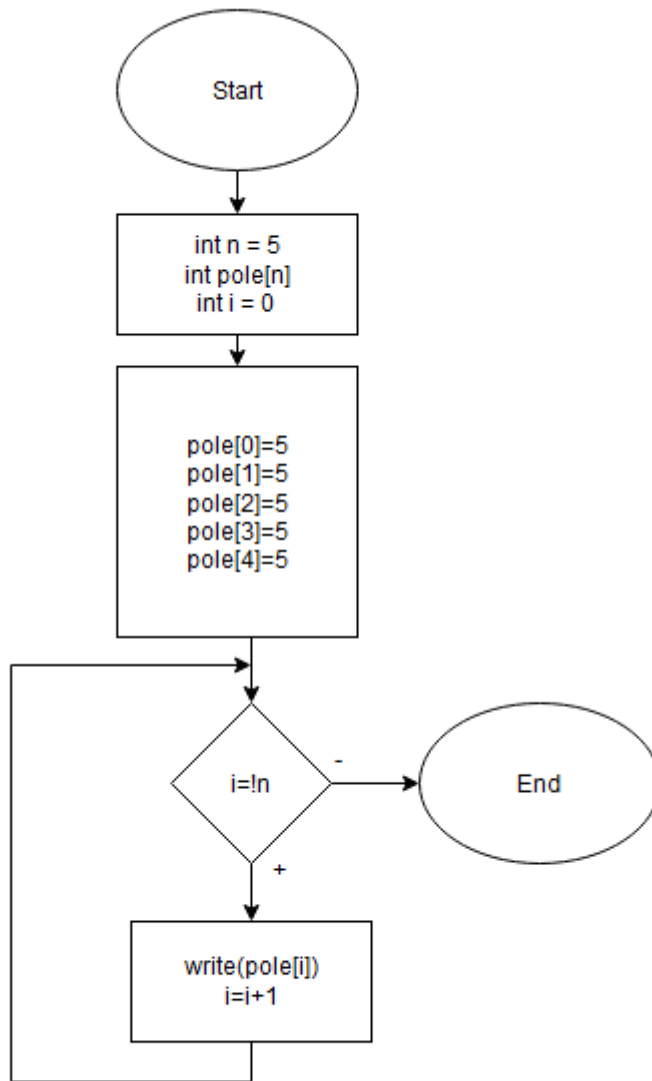
(pokud zbyde čas, tak žáci udělají i ostatní cykly)

Cyklus Do-while:

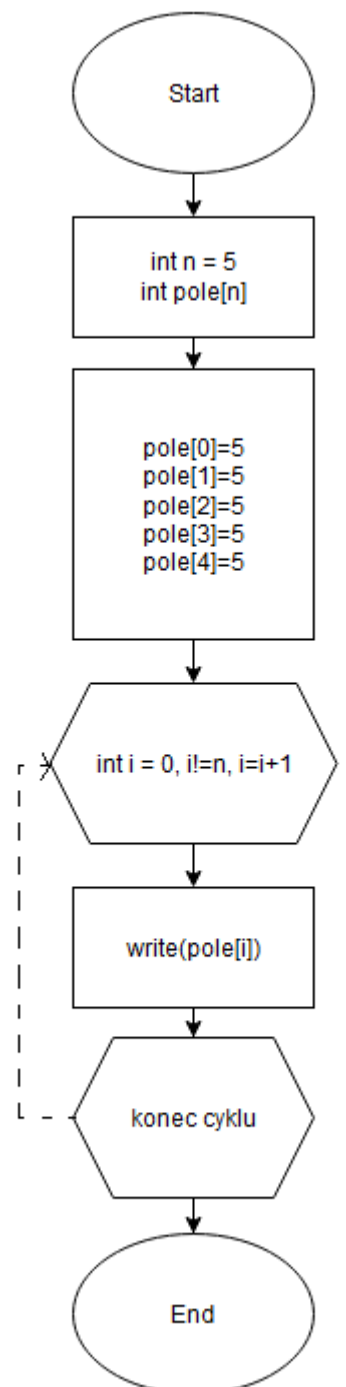


Cyklus

While:



Cyklus For:



9.

Téma vyučovací hodiny:**Třídící algoritmy – Bubble sort****Cíle vyučovací hodiny:****Zopakovat, simulovat****Organizační forma vyučovací hodiny:****Frontální výuka**

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci		
5	Administrace		Třídící algoritmy – Bubble sort	Počítače Projektor Interaktivní tabule
10	Prezentace na téma třídících algoritmů – Bubble sort	Naslouchání prezentaci a tvorba poznámek		
15				
20				
25	Společná tvorba vývojového diagramu (nejdříve studentské nápady)	Společná tvorba vývojového diagramu		
30				
35				
40	Dotazy	Kladení dotazů a nejasností		
45	Shrnutí			

Tabulka 11 – Vyučovací hodina: Třídící algoritmy – Bubble sort

Popis vyučovací hodiny:

Tématem této vyučovací hodiny třídící algoritmy, konkrétně se zde zaměříme na Bubble sort.

Odborným obsahem v této hodině jsou třídící algoritmy.

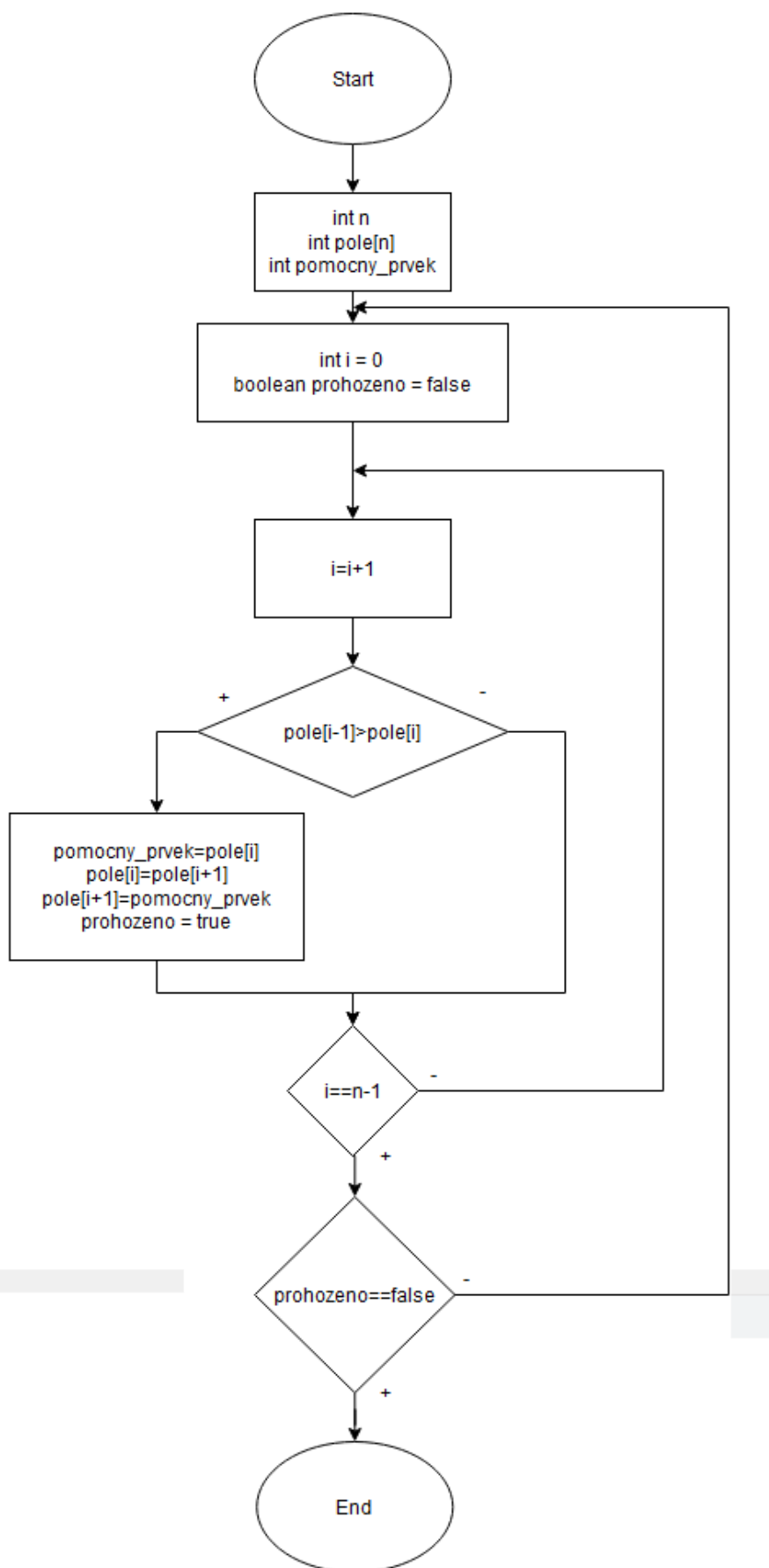
Žáci se tuto hodinu naučí zopakovat funkci třídících algoritmů a nasimulovat funkci Bubble sortu.

Celá vyučovací hodina proběhne ve formě frontální výuky s možnostmi dotazů od žáků a na konec proběhne společná tvorba vývojového diagramu Bubble sortu.

Pokyny pro vyučujícího:

Vyučující využívá pro výuku prezentaci s názvem: 9_tridici_algoritmy.pptx

Vývojový diagram Bubble sortu se nachází na následující straně.



10.

Téma vyučovací hodiny:**Algoritmizace - shrnutí****Cíle vyučovací hodiny:****Zopakovat, vyjádřit****Organizační forma vyučovací hodiny:****Frontální výuka**

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci	Algoritmizace	Počítače Projektor Interaktivní tabule
5	Administrace			
10	Prezentace shrnující celou dosavadní kapitolu - Algoritmizace	Naslouchání prezentaci a tvorba poznámek		
15				
20				
25				
30				
35	Procvičování učiva, které není jasné resp. opakování tvorby složitějších algoritmů	Procvičování učiva, které není jasné resp. opakování tvorby složitějších algoritmů		
40				
45				

Tabulka 12 – Vyučovací hodina: Algoritmizace - shrnutí

Popis vyučovací hodiny:

Tématem této vyučovací hodiny je shrnutí celé kapitoly algoritmizace.

Odborným obsahem v této hodině je algoritmizace.

Žáci si tuto hodinu zopakují, co se doposud naučili v kapitole algoritmizace a budou to umět vyjádřit vlastními slovy.

Vyučovací hodina proběhne nejprve ve formě frontální výuky s možnostmi dotazů od žáků a na závěr bude možnost pro zopakování nejasností z celé kapitoly.

Pokyny pro vyučujícího:

Vyučující využívá pro výuku prezentaci s názvem: 10_algoritmizac_shrnuti.pptx

Pokud žáci nebudou chtít zopakovat nic konkrétního, tak použít nějaký algoritmus z předchozích hodin.

11.

Téma vyučovací hodiny:**Test - algoritmizace, Úvod do programování****Cíle vyučovací hodiny:****Prokázat, vysvětlit, popsat****Organizační forma vyučovací hodiny:****Frontální výuka + samostatná práce**

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci		
5	Administrace			Počítače Projektor Interaktivní tabule
10	Test - Algoritmizace	Vyplňování testu	Algoritmy	
15				
20				
25				
30				
35	Prezentace na téma úvod do programování	Naslouchání prezentaci a tvorba poznámek	Úvod do programování	
40				
45				

Tabulka 13 – Vyučovací hodina: Test – Algoritmizace, Úvod do programování

Popis vyučovací hodiny:

Tématem této vyučovací hodiny je ověřovací test shrnujícího kapitoly algoritmy. Ve zbytku času úvod do programování.

Odborným obsahem probíraným v této hodině jsou algoritmy a úvod do programování.

Žáci tuto hodinu prokáží znalosti z oblasti algoritmy, dokážou vysvětlit tvorbu algoritmu. Ve zbytku času se naučí popsat základní problematiku úvodu do programování.

Nejprve začíná hodina samostatnou prací – testem, poté proběhne frontální výuka úvodu do programování.

Pokyny pro vyučujícího:

Test se nachází v souboru s názvem test_algoritmy.docx, který obsahuje 2 varianty. Správné odpovědi se nachází v souboru s názvem test_algoritmy_odpovedi.docx.

Vyučující využívá pro výuku prezentaci s názvem: 11_uvod_do_programovani.pptx

12.

Téma vyučovací hodiny:**Python - úvod****Cíle vyučovací hodiny:****Definovat, popsat****Organizační forma vyučovací hodiny:****Frontální výuka**

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci	Python	Počítače Projektor Interaktivní tabule
5	Administrace			
10	Prezentace na téma Python	Naslouchání prezentaci a tvorba poznámek		
15				
20	Popis vývojového prostředí	Vyzkoušení práce s IDLE		
25				
30				
35	Dotazy	Kladení dotazů a nejasností		
40	Shrnutí			
45				

Tabulka 14 – Vyučovací hodina: Python - úvod

Popis vyučovací hodiny:

Tématem této vyučovací hodiny je úvod do jazyka Python,

Odborným obsahem v této hodině je Python.

Žáci se tuto hodinu naučí definovat a popsat základní vlastnosti jazyka Python.

Celá vyučovací hodina proběhne ve formě frontální, během které si žáci na počítačích zkouší tvorbu jednoduchých programků a práci v novém programovacím prostředí.

Pokyny pro vyučujícího:

Vyučující využívá pro výuku prezentaci s názvem: 12_python_uvod.pptx

Vyučující využívá pro výuku dokument s názvem: idle_navod.docx

Příklady pro poznání výukového prostředí:

```
>>> 6*6           násobení
36
```

```
>>> a=5           zápis proměnné a,b,c a tisk c
>>> b=5
>>> c=(a+b)
>>> print(c)
10
```

```
>>> 2**4          Umocňování (číslo ** příslušná mocnina)
16
```

```
>>> print('ahoj') tisk řetězce
ahoj
```

```
>>> import math   importování knihovny math pro využití funkce sqrt (druhá odmoc-
>>> math.sqrt(9)  nina)
3.0
```

13.

Téma vyučovací hodiny:

Python – první programování

Cíle vyučovací hodiny:

Vyzkoušet

Organizační forma vyučovací hodiny:*Frontální výuka + samostatná práce*

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci		
5	Administrace			Počítače Projektor Interaktivní tabule
10	Praktické ukázky zápisu programu v Pythonu	Psaní programu v Pythonu	Python	
15				
20				
25				
30				
35				
40	Dotazy	Kladení dotazů a nejasností		
45				

Tabulka 15 – Vyučovací hodina: Python – první programování

Popis vyučovací hodiny:

Tématem této vyučovací je tvorba prvních programů v Pythonu.

Odborným obsahem v této hodině je Python.

Žáci si tuto hodinu vyzkouší tvorbu prvních programů v jazyce Python.

Celá vyučovací hodina proběhne ve formě frontální výuky, během které si žáci zkusí tvorbu programů i u sebe na počítačích.

Pokyny pro vyučujícího:

Úkol: Vytvořte program, do kterého zadáte číselnou hodnotu proměnné ,a' a ,b' a program vyhodnotí, jestli je $a > b$ nebo $a == b$ nebo $a < b$ a tuto skutečnost vytiskne.

Řešení:

```
a = 4
b = 5
```

```
if (a>b):
    print('a je vetsi nez b')
elif (a==b):
    print('a je stejne jako b')
else:
    print('b je vetsi nez a')
```

b je vetsi nez a

```
a = 5
b = 5
```

```
if (a>b):
    print('a je vetsi nez b')
elif (a==b):
    print('a je stejne jako b')
else:
    print('b je vetsi nez a')
```

a je stejne jako b

```
a = 6
b = 5
```

```
if (a>b):
    print('a je vetsi nez b')
elif (a==b):
    print('a je stejne jako b')
else:
    print('b je vetsi nez a')
```

a je vetsi nez b

Úkol: Vytvořte program, který vypočítá z hodnoty proměnné a, její druhou, třetí, čtvrtou a pátou mocninou a druhou odmocninou a všechny tyto hodnoty vypíše.

Řešení:

```
import math

a = 4

print(a**2)          16
print(a**3)          64
print(a**4)          256
print(a**5)          1024
print(a**5)          2.0
print(math.sqrt(a)) >>>
```

Úkol: Vytvořte program, kde zadáte hodnotu proměnné a. Poté se porovná, zda druhá mocnina odmocniny z a se rovná a. Pokud ano, vytiskne ano, pokud ne, vytiskne ne.

Řešení:

```
import math

a = 5

if (math.sqrt(a)**2)==a):
    print('ano')
else:
    print('ne')
    ne
    >>>

import math

a = 4

if (math.sqrt(a)**2)==a):
    print('ano')
    ano
else:
    print('ne')
    >>>
```

14.

Téma vyučovací hodiny:

Python – zápis proměnných, cyklů

Cíle vyučovací hodiny:

Vyzkoušet, aplikovat

Organizační forma vyučovací hodiny:*Frontální výuka + samostatná práce*

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci	Python	Počítače Projektor Interaktivní tabule
5	Administrace			
10	Prezentování prezentace na téma zápisu proměnných a cyklů v Pythonu	Kladení dotazů a nejasností		
15				
20				
25	Samostatná práce	Vypracovávání samostatné práce		
30				
35				
40	Zhodnocení samostatné práce			
45				

Tabulka 16 – Vyučovací hodina: Python – zápis proměnných, cyklů

Popis vyučovací hodiny:

Tématem této vyučovací je zápis proměnných a cyklů v jazyce Python.

Odborným obsahem v této hodině je Python.

Žáci si tuto hodinu vyzkouší tvorbu složitějších programů obsahující proměnné a cykly, naučí se aplikovat nabyté znalosti z oblasti algoritmizace.

Vyučovací hodina začne ve formě frontální výuky, během které si žáci zkouší tvorbu programů i u sebe na počítačích. Poté je zadána samostatná práce, která je posléze zkontrolována.

Pokyny pro vyučujícího:

Vyučující využívá pro výuku prezentaci s názvem: 14_python_promenne_cykly

Úkol: Za pomoci cyklu vytiskněte 5x hlášku ahoj, za kterou bude následovat číslice, postupně od 0 do 4.

Řešení:

!V základní verzi Pythonu se nenachází cyklus Do-While (Repeat-Until)!

Cyklus While:

```
i=0                                ahoj 0
while (i!=5):                       ahoj 1
    print('ahoj', i)                 ahoj 2
    i=i+1                             ahoj 3
                                      ahoj 4
```

Cyklus For:

```
for x in range(0,5,1):              ahoj 0
    print('ahoj', x)                 ahoj 1
                                      ahoj 2
                                      ahoj 3
                                      ahoj 4
```

Úkol: Vytiskněte za pomoci cyklu For všechna lichá čísla od 1 do 50.

Řešení:

```
for x in range(1, 50, 2):  
    print(x)  
1  
3  
5  
7  
9  
11  
13  
15  
17  
19  
21  
23  
25  
27  
29  
31  
33  
35  
37  
39  
41  
43  
45  
47  
49
```

15.

Téma vyučovací hodiny:**Python – tvorba jednoduchých programů****Cíle vyučovací hodiny:****Řešit****Organizační forma vyučovací hodiny:*****Samostatná práce***

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci		
5	Administrace			Počítače Projektor Interaktivní tabule
10	Zadání samostatné práce	Práce na samostatné práci	Python	
15				
20	Kontrola samostatné práce	Prezentace výsledků		
25	Zadání samostatné práce	Práce na samostatné práci		
30				
35				
40	Kontrola samostatné práce	Prezentace výsledků		
45				

Tabulka 17 – Vyučovací hodina: Python – tvorba jednoduchých programů

Popis vyučovací hodiny:

Tématem této vyučovací je tvorba jednoduchých programů v jazyce Python...

Odborným obsahem v této hodině je Python.

Žáci budou tuto hodinu samostatně řešit zadané úlohy.

Vyučovací hodina proběhne ve formě samostatné práce, během které žáci samostatně řeší zadané úkoly, které jsou poté hromadně zkontrolovány.

Pokyny pro vyučujícího:

Vyučující využívá pro výuku prezentaci s názvem:

Úkol: Vytvořte program, kde se zadá poloměr kruhu a program z něj vypočítá obvod a obsah kruhu a tyto hodnoty vypíše.

Řešení:

```
import math

print('zadejte polomer kruhu')
r = float(input())
print('obvod kruhu je: ', (2*math.pi*r))
print('obsah kruhu je: ', (math.pi*r**2))

zadejte polomer kruhu
10
obvod kruhu je: 62.83185307179586
obsah kruhu je: 314.1592653589793
```

Úkol: Vytvořte program, kde se zadá libovolné číslo a program vypočítá jeho třetí mocninu. Pokud bude tato třetí mocnina sudá, vypíše, že je sudá, pokud bude lichá, vypíše tuto odmocninu 5x za sebou.

Řešení:

```
print('zadejte cislo')
i = float(input())
i=i**3

if i%2==0:
    print('je sude')
else:
    for x in range(0,5,1):
        print(i)

zadejte cislo
2
je sude

zadejte cislo
3
27.0
27.0
27.0
27.0
27.0
```

16.

Téma vyučovací hodiny:**Python – přepisování vývojových diagramů****Cíle vyučovací hodiny:****Přeformulovat, prokázat****Organizační forma vyučovací hodiny:****Frontální výuka + samostatná práce**

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci	Python	Počítače Projektor Interaktivní tabule
5	Administrace			
10	Vzorová ukázka přepisování vývojových diagramů do Pythonu	Naslouchání prezentaci a psaní programu v Pythonu		
15				
20				
25	Samostatná práce	Práce na samostatné práci		
30				
35				
40	Zhodnocení samostatné práce	Prezentace výsledků		
45				

Tabulka 18 – Vyučovací hodina: Python – přepisování vývojových diagramů

Popis vyučovací hodiny:

Tématem této vyučovací je přepis vývojových diagramů do jazyce Python.

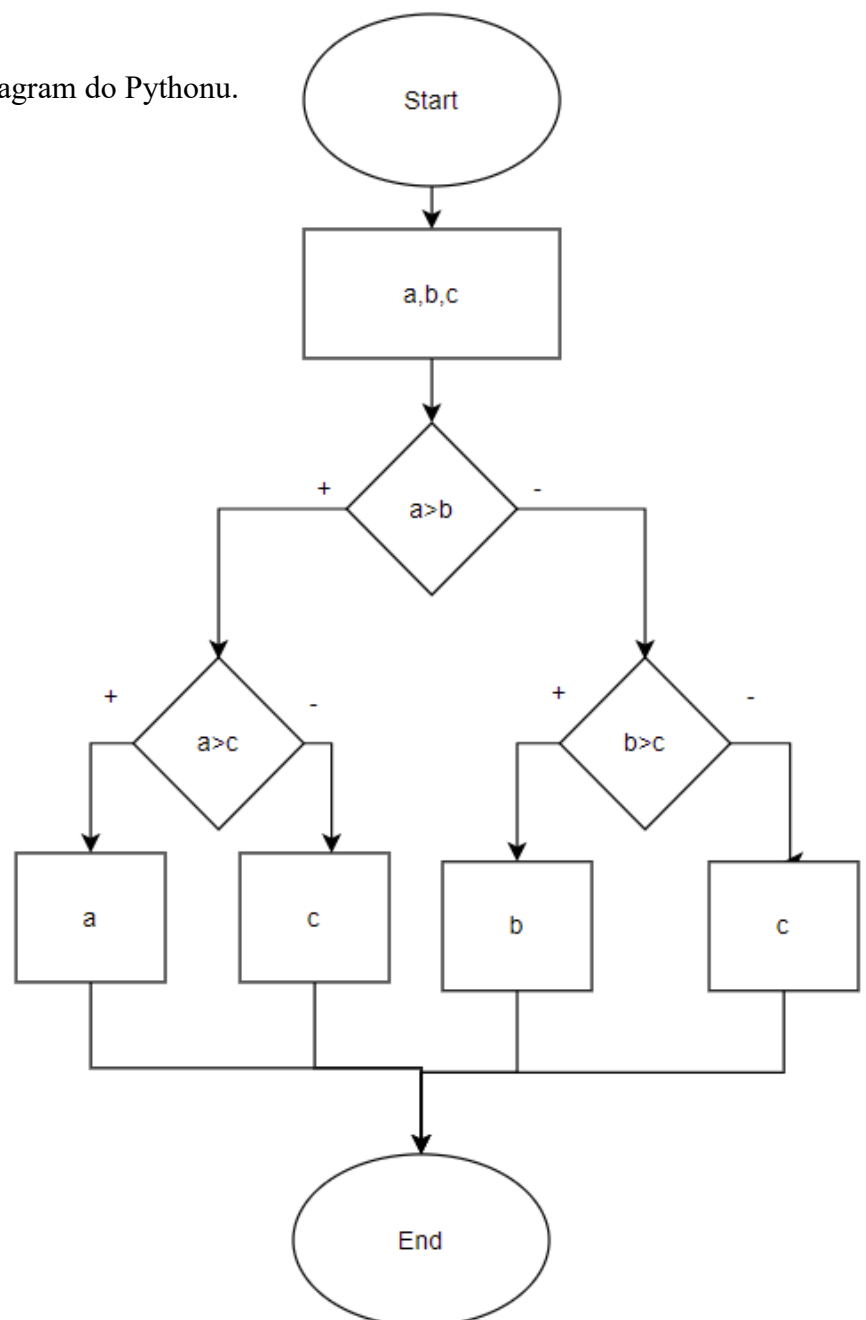
Odborným obsahem v této hodině je Python.

Žáci si tuto hodinu vyzkouší a prokáží schopnost přeformulovat učivo z minulé kapitoly – algoritmizace, do nynější kapitoly – programování.

Vyučovací hodina začne ve formě frontální výuky, během které vyučující ukáže způsob přepisu vývojových diagramů do jazyk Python. Poté následuje samostatná práce, kde žáci sami tento postup aplikují.

Pokyny pro vyučujícího:

Úkol: Přepište následující diagram do Pythonu.



Řešení:

```
print ('zadejte a')
a=int (input ())

print ('zadejte b')
b=int (input ())

print ('zadejte c')
c=int (input ())
```

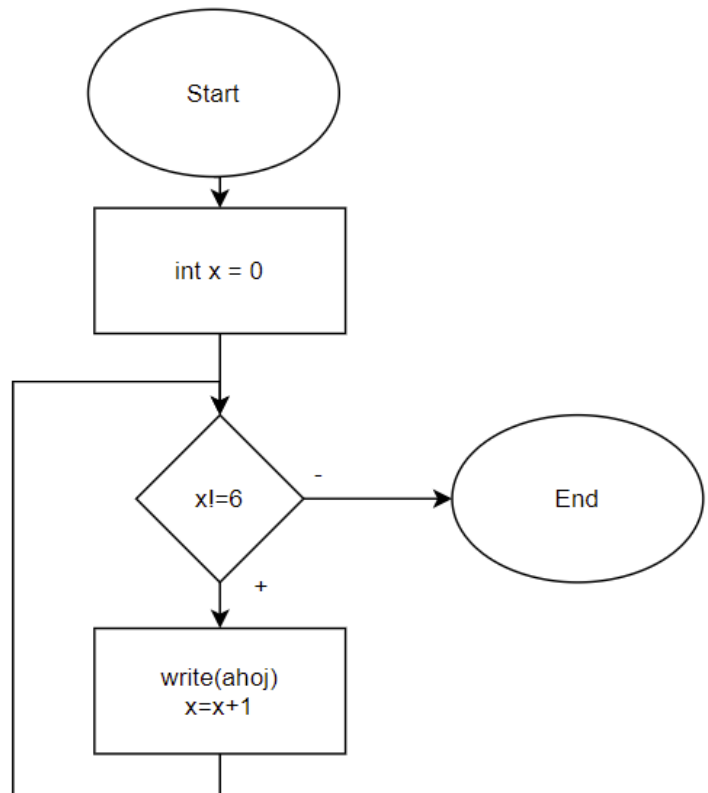
```
if (a>b) :
    if (a>c) :
        print ('a')
    else :
        print ('c')
else :
    if (b>c) :
        print ('b')
    else :
        print ('c')
```

```
zadejte a
1
zadejte b
2
zadejte c
3
c
```

```
zadejte a
3
zadejte b
2
zadejte c
1
a
```

```
zadejte a
1
zadejte b
3
zadejte c
2
b
```

Zadání: Přepište následující diagram do Pythonu.



Řešení:

```
x=0
while (x!=6) :
    print ('ahoj')
    x=x+1
    ahoj
    ahoj
    ahoj
    ahoj
    ahoj
    ahoj
```


17.

Téma vyučovací hodiny:**Python – tvorba komplexního programu I.****Cíle vyučovací hodiny:****Načrtnout, kombinovat****Organizační forma vyučovací hodiny:*****Samostatná práce***

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci		
5	Administrace			Počítače Projektor Interaktivní tabule
10	Samostatná práce (tvorba vývojového diagramu + přepis)	Práce na samostatné práci	Python	
15				
20				
25				
30				
35	Zhodnocení samostatné práce	Prezentace výsledků		
40				
45				

Tabulka 19 – Vyučovací hodina: Python – tvorba komplexního programu I.

Popis vyučovací hodiny:

Tématem této vyučovací hodiny je tvorba složitějších programů v jazyce Python.

Odborným obsahem v této hodině je Python.

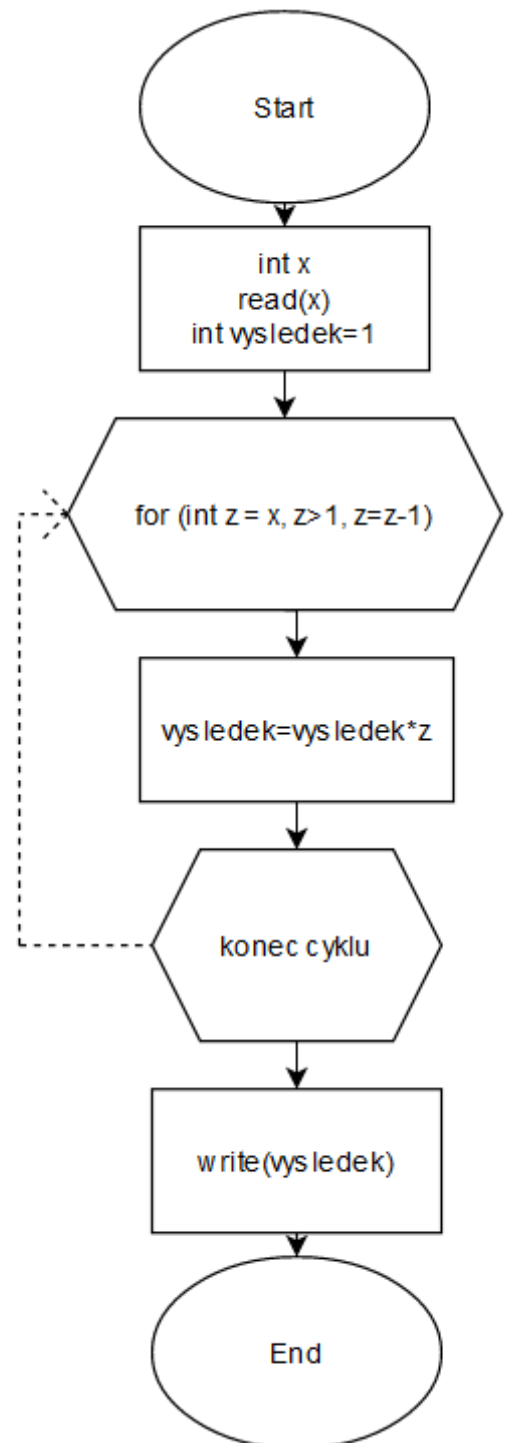
Žáci si tuto hodinu vyzkouší načrtnout postup komplexního vytváření programů (vývojový diagram + samostatný kód), tedy zkombinovat dříve nabyté vlastnosti.

Vyučovací hodina je ve formě samostatné práce, která je na konci hodiny společně zkontrolována.

Pokyny pro vyučujícího:

Zadání: Vytvořte vývojový diagram a poté program, který bude sloužit k výpočtu faktoriálu zadaného čísla. *Bonus: V pythonu ověřte Vámi vypočtený faktoriál za pomoci funkce `math.factorial` a pokud budou oba výsledky stejné, vypíše se hláška, že jsou stejné, pokud budou rozdílné, vypíše se hláška, že jsou rozdílné a bude nutno najít chybu.*

Řešení:



```
print('zadejte cislo')
x = int(input())
vysledek = 1

for z in range(x,1,-1):
    vysledek=vysledek*z

print('vypocitany vysledek:', vysledek)

import math

print('vysledek pomoci funkce:', math.factorial(x))

if(vysledek == math.factorial(x)):
    print('vysledky jsou stejne')
else:
    print('vysledky jsou rozdilne')

zadejte cislo
5
vypocitany vysledek: 120
vysledek pomoci funkce: 120
vysledky jsou stejne
```

18.

Téma vyučovací hodiny: Python – tvorba komplexního programů II.
Cíle vyučovací hodiny: Načrtnout, kombinovat
Organizační forma vyučovací hodiny: *Samostatná práce*

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci		
5	Administrace			Počítače Projektor Interaktivní tabule
10	Samostatná práce (tvorba vývojového diagramu + přepis)	Práce na samostatné práci	Python	
15				
20				
25				
30				
35	Zhodnocení samostatné práce	Prezentace výsledků		
40				
45				

Tabulka 20 – Vyučovací hodina: Python – tvorba komplexního programu II.

Popis vyučovací hodiny:

Tématem této vyučovací hodiny je tvorba složitějších programů v jazyce Python, další část.

Odborným obsahem v této hodině je Python.

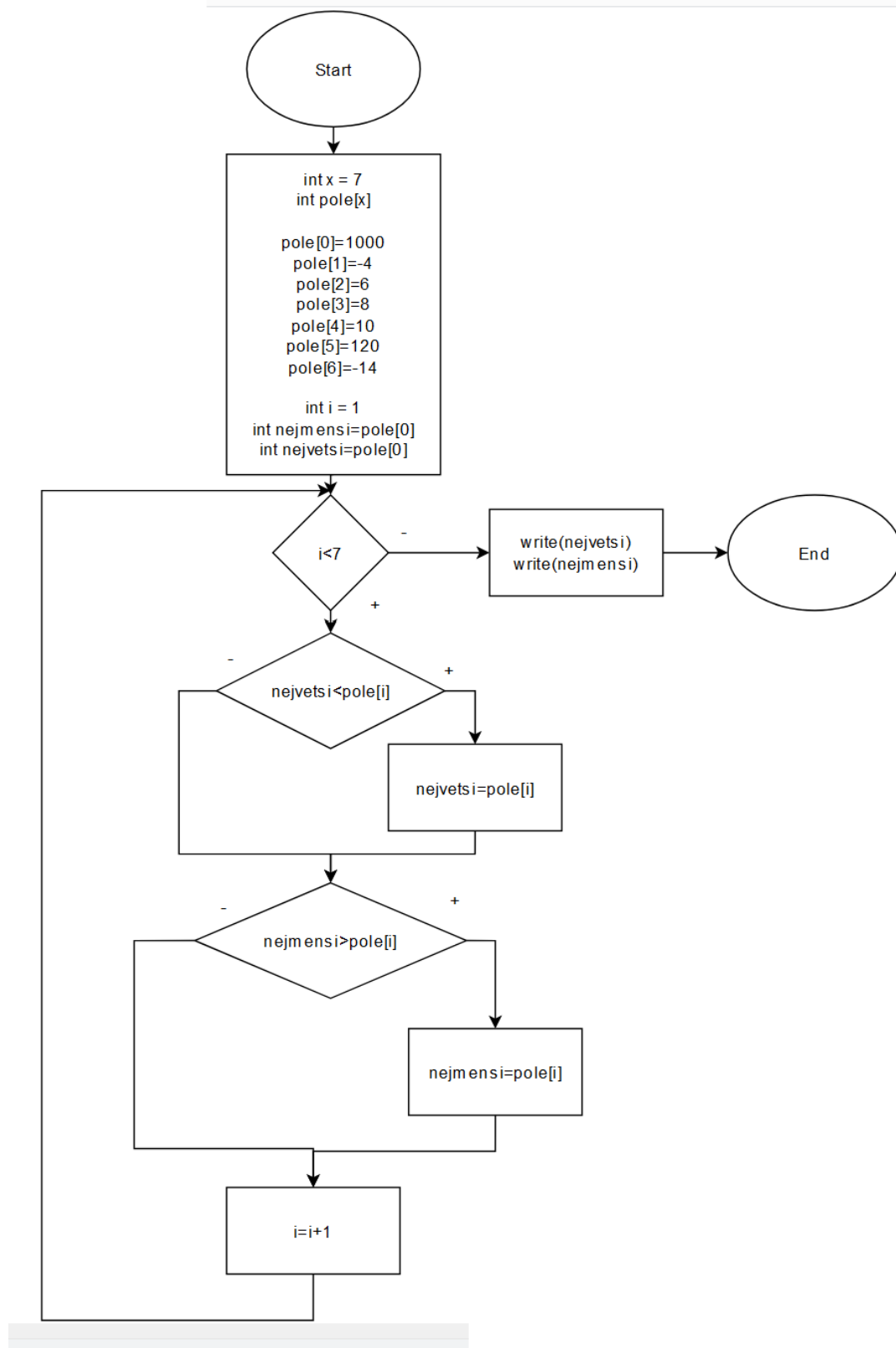
Žáci si tuto hodinu vyzkouší načrtnout postup komplexního vytváření programů (vývojový diagram + samostatný kód), tedy kombinovat a aplikovat dříve nabyté vědomosti.

Vyučovací hodina je ve formě samostatné práce, která je na konci hodiny společně zkontrolována.

Pokyny pro vyučujícího:

Zadání: Vytvořte vývojový diagram a napište program, který v zadaném poli najde největší a nejmenší prvek. Tyto prvky vytiskne. *Bonus: na závěr vypište obsah pole.*

Řešení:



```
import array as arr
pole = arr.array('i', [1000, -4, 6, 8, 10, 120, -14])

i=1
nejvetsi=pole[0]
nejmensi=pole[0]
while (i<7):
    if (nejvetsi<pole[i]):
        nejvetsi=pole[i]
    if(nejmensi>pole[i]):
        nejmensi=pole[i]
    i=i+1

print('nejvetsi:', nejvetsi)
print('nejmensi:', nejmensi)

for x in range (0,7,1):
    print (pole[x])
```

Bonus:

Zadání: Vytvořte program, který za pomoci 3 vnořených cyklů for vypíše čísla od 0 (respektive 000) po 999.

Řešení:

```
for x in range(0,10,1):
    for y in range(0,10,1):
        for z in range(0,10,1):
            print(x,y,z)
```

9 6 2
9 6 3
9 6 4
9 6 5
9 6 6
9 6 7
9 6 8
9 6 9
9 7 0
9 7 1
9 7 2
9 7 3
9 7 4
9 7 5
9 7 6
9 7 7
9 7 8
9 7 9
9 8 0
9 8 1
9 8 2
9 8 3
9 8 4
9 8 5
9 8 6
9 8 7
9 8 8
9 8 9
9 9 0
9 9 1
9 9 2
9 9 3
9 9 4
9 9 5
9 9 6
9 9 7
9 9 8
9 9 9

19.

Téma vyučovací hodiny:**Test - Python****Cíle vyučovací hodiny:****Prokázat****Organizační forma vyučovací hodiny:*****Samostatná práce***

Časový průběh a organizace hodiny (použité metody a formy práce, motivace, prověřování vědomostí, nové pojmy)			Odborný obsah vyučovaného tématu	Použité pomůcky, didaktická technika, literatura
čas	Učitel	Žáci		
5	Administrace			Počítače Projektor Interaktivní tabule
10	Zadání testu	Vyplňování testu	Python	
15				
20				
25				
30				
35				
40	Dotazy	Kladení dotazů a nejasností		
45				

Tabulka 21 – Vyučovací hodina: Test - Python

Popis vyučovací hodiny:

Tématem této vyučovací hodiny je test znalostí z Pythonu.

Odborným obsahem v této hodině je Python.

Žáci v testu prokáží znalost základů jazyka Python.

Vyučovací hodina je ve formě samostatné práce, tedy samostatného vyplňování. Vývojový diagram je možno psát rukou nebo na počítači, taktéž program lze, dle uvážení žáka, psát na počítači nebo na papír.

Pokyny pro vyučujícího:

Test se nachází v souboru s názvem test_python.docx, který obsahuje 2 varianty.

Správné odpovědi se nachází v souboru s názvem test_python_odpovedi.docx.

6 DOTAZNÍKOVÉ OVĚŘENÍ VÝUKOVÝCH MATERIÁLŮ

Jako ověření funkčnosti výukových materiálů, dle zadání, byl vypracován dotazník. Respondenti na něj měli možnost odpovídat v online formě přes nástroj Google Forms. Vzhledem ke komplexnosti dotazníku, z něj plynula nutnost velmi důkladně projít jednotlivé materiály. To bylo pro respondenty – učitele z praxe časově velmi náročné, ale někteří i tak nad rámec dotazníku, poskytli zpětnou vazbu i v písemné formě.

Celý dotazník je uveden na následujících stranách, konkrétně od stany 79 do strany 82.

Bc. Martin Majer

Dotazník výukových materiálů do Informatiky v rámci diplomové práce

Vážené respondentky, vážení respondenti,

obracím se na Vás s žádostí o vyplnění mého dotazníku, který poslouží jako zpětná vazba podkladů pro výuku Algoritmizace a programování v rámci mé Diplomové práce.

Dotazník je anonymní.

Vaše odpovědi, prosím, zakroužkujte. Pokud na danou otázku nechcete odpovídat, nic nekroužkujte, ani nic nikam nepište a danou otázku přeskočte.

Předem děkuji za Váš čas.

1. Jak celkově hodnotíte výukové materiály?

výborné chvalitebné dobré dostatečné nedostatečné

2. Jak hodnotíte samostatně výukové prezentace?

výborné chvalitebné dobré dostatečné nedostatečné

3. Jak hodnotíte samostatně textové materiály?

výborné chvalitebné dobré dostatečné nedostatečné

4. Jak hodnotíte samostatně testy?

výborné chvalitebné dobré dostatečné nedostatečné

5. Jak hodnotíte část algoritmizace?

výborné chvalitebné dobré dostatečné nedostatečné

7. Dovedete si představit, že byste podle těchto materiálů vyučovali?

určitě ano ano, ale s výhradami nevím spíše ne určitě ne

8. Jak hodnotíte gramatickou správnost materiálů?

výborné chvalitebné dobré dostatečné nedostatečné

9. Jak hodnotíte využití pomůcek a didaktické techniky?

výborné chvalitebné dobré dostatečné nedostatečné

10. Shledáváte v materiálech využívání různorodých metod a forem práce?

určitě ano spíše ano nevím spíše ne určitě ne

11. Shledáváte zvolené časové rozvržení vyučovacích hodin jako vhodné?

určitě ano spíše ano nevím spíše ne určitě ne

12. Jak hodnotíte organizaci jednotlivých hodin?

výborné chvalitebné dobré dostatečné nedostatečné

13. Jak hodnotíte grafické zpracování materiálů?

výborné chvalitebné dobré dostatečné nedostatečné

14. Jak hodnotíte formální zpracování materiálů?

výborné chvalitebné dobré dostatečné nedostatečné

15. Je podle Vás hloubka znalostí předávaná ve výukových materiálech dostatečná?

určitě ano spíše ano nevím spíše ne určitě ne

16. Je ve výukových materiálech něco konkrétního, co byste pochválil/a, nebo naopak zkritizoval/a? (pokud ne, neodpovídejte)

.....

.....

.....

.....

17. Chybí ve výukových materiálech podle Vás něco důležitého, popřípadě co byste udělali jinak? (pokud nic, neodpovídejte)

.....

.....

.....

.....

18. Pokud máte k materiálům jakýkoliv doplňující komentář, napište ho prosím zde. (pokud ne, nepište nic)

.....

.....

.....

.....

.....

.....

.....

Děkuji, za vyplnění dotazníku. Bc. Martin Majer

6.1 Vyhodnocení dotazníku a zpětné vazby

I přes to, že jsem ke sběru informací použil dotazník, což jsem typická součást kvantitativního výzkumu, považuji svůj výzkum spíše za kvalitativní.

Vzhledem k malému počtu oslovených učitelů a jejich ochotě podrobit mé podklady důkladnému zkoumání a následně nejen vyplnit dotazník, ale i doplnit tento dotazník rozsáhlým textovým hodnocením či dokonce osobní konzultací, usuzuji, že výzkum probíhal mnohem více do hloubky, než když bych jen statisticky vyhodnocoval dotazníky od velkého počtu respondentů. Proto zde nemá smysl uvádět různé grafy, kde budou vyobrazeny četnosti odpovědí na danou otázku, jelikož pro tak malý vzorek navrátivších se dat, by to ze statistického pohledu, stejně postrádalo výpovědní hodnotu.

Abych to shrnul, výzkum i přes použití dotazníku šel spíše do hloubky, tedy jedná se spíše o kvalitativní výzkum, než povrchní kvantitativní výzkum. Cílem bylo získat kvalitní a použitelnou zpětnou.

Na základě komplexní zpětné vazby jsem proto mohl mnohem lépe upravit výukové materiály do konečné podoby.

Nemá smysl zde uvádět kompletní zpětnou vazbu od učitelů, jelikož v mnohém odkazují na problémy, které se vyskytovaly v rámci dřívějších verzí prezentací, ve verzích, které jsou uvedeny v příloze P3, již tyto chyby nenajdeme.

Nejvíce jsem rád za to, že se respondenti podívali na mé materiály z opačné strany. Ze strany středoškolských pedagogů. Vysokoškolský a středoškolský přístup se v mnohém rozcházel. Uvedu jeden příklad: z vysokoškolského pohledu je nutno, převzaté citace/obrázky, vždy exaktně citovat, takže například u obrázků se vyskytují čísla v hranatých závorkách, které odkazují na seznam citací. Ve středoškolském pohledu působí tato čísla v hranatých závorkách rušivě a je to věc, na kterou žáci zbytečně upínají svoji pozornost.

Hlavní věcí, kterou jsem na základě zpětné vazby upravoval, byly prezentace. Jednak jsem do některých vložil ilustrační obrázky, další prezentace jsem doplnil o „animace“ a především jsem zapracoval na stylistických záležitostech textů. Nejvíce mi v tomto ohledu pomohl pohled jednoho z respondentů, se kterým jsem na osobní konzultaci všechny prezentace společně procházel a diskutoval nad jednotlivými slajdy.

I přes všechny nedokonalosti v prvotních verzích výukových materiálů, byla zpětná vazba vesměs pozitivní a vyučující by si uměli představit vycházet z mých výukových materiálů, i když by se na nich ještě nějaká práce musela udělat. To by ale ukázala až praxe.

Těžko je materiály hodnotit pouze teoreticky. Pro skutečné otestování kvality výukových materiálů by bylo nutno je podrobit zátěžovým testům v rámci reálného vyučování.

Prezentace, které se nacházejí na CD v příloze P3, jsou již po úpravách na základě informací obsažených ve zpětné vazbě od středoškolských učitelů.

ZÁVĚR

Nejprve jsem se seznámil s RVP pro gymnázia a pro obor Informační technologie, ze kterých jsem vycházel při tvorbě výukových materiálů. Výukové materiály jsem posléze rozdělil na dvě části. Část zabývající se výukou algoritmizace a na část zabývající se výukou základů programování.

Na základě provedeného výzkumu v rámci středních škol a gymnázií jsem vyhodnotil, že nejpoužívanějším jazykem pro výuku programování je jazyk Python. Toto zjištění nebylo úplně velkým překvapením. Jazyk Python se těší v dnešní době veliké oblibě, jednak u laické veřejnosti a jednak v profesionální branži.

K výuce programování jsem zvolil vývojové prostředí IDLE z důvodu jednoduchosti pro nové uživatele a také kvůli tomu, že po instalaci Pythonu je toto prostředí již nainstalováno.

Dále jsem vytvořil konkrétní výukové materiály. Nejprve jsem vytvořil průvodní dokument, kde jsou obsaženy obecné náležitosti jako cíle, klíčové kompetence atd. Dále tento dokument obsahuje konkrétní časový popis jednotlivých hodin, popis činností pro vyučujícího nebo třeba konkrétní řešené úlohy. K tomu byly vytvořeny výukové podkladové prezentace, konkrétně celkem 10, kde se z nich 7 zabývají oblastí algoritmizace a zbylé 3, se zabírají oblastí programování, konkrétně programovacím jazykem Python.

V rámci výukových materiálů ještě vznikly 2 sady testových úloh. Každá obsahuje 2 verze, pro 2 skupiny žáků, jeden test na znalosti nabyté v oblasti algoritmizace a ten druhý slouží k testování znalostí programování.

Poslední součástí výukových materiálů je návod pro práci v prostředí IDLE, kde poskytují obrázek o práci v prostředí i pro úplně začátečníky. Vše je doplněno snímkem z konkrétně používaného prostředí.

Na závěr byly vypracované výukové materiály zaslány vyučujícím z praxe, učitelům vyučujícím na středních školách a gymnáziích, kde po prostudování výukových materiálů, tito vyučující poskytli skrze online dotazník zpětnou vazbu, na základě které byly dodatečně upraveny výukové materiály s ohledem na zkušenosti učitelů z praxe.

Bohužel se potvrdily obavy, že prostudování výukových materiálů učiteli z praxe je velmi časově náročné a není možno získat tolik zpětné vazby. Zúčastněným učitelům děkuji za zpětnou vazbu a vážím si jejich poskytnutého času.

SEZNAM POUŽITÉ LITERATURY

- [1] *RÁMCOVÉ VZDĚLÁVACÍ PROGRAMY* [online]. [cit. 2021-02-11]. Dostupné z: <http://www.nuv.cz/t/rvp>
- [2] BALADA, Jan. *Rámcový vzdělávací program pro gymnázia: RVP G*. Praha: Výzkumný ústav pedagogický v Praze, c2007. ISBN 978-80-87000-11-3.
- [3] *Rámcový vzdělávací program pro obor vzdělání 18-20-M/01 Informační technologie*. 2008. Praha: Ministerstvo školství, mládeže a tělovýchovy.
- [4] *Co je informatické myšlení?* [online]. [cit. 2021-03-02]. Dostupné z: <https://www.imysleni.cz/informaticke-mysleni/co-je-informaticke-mysleni>
- [5] HYLMAR, Radek. *Programování pro úplné začátečníky*. Brno: Computer Press, 2009. ISBN 978-80-251-2129-0.]
- [6] LANDA, L. N. *Algoritmy a učení: Kybernetika, algoritmizace a heuristika ve vyučování*. Praha: Státní pedagogické nakladatelství, n. p., 1973.
- [7] KMOCH, Petr. *Informatika a výpočetní technika pro střední školy*. Aktualiz. vyd. Brno: Computer Press, 2004. Učebnice pro střední školy (Computer Press). ISBN 80-251-0376-5.
- [8] VITÁSEK, Tomáš. *Programovací jazyk Python a účelnost jeho zařazení do výuky*. 2010. Bakalářská práce. Vysoká škola ekonomická v Praze.
- [9] *History and License* [online]. [cit. 2021-5-6]. Dostupné z: <https://docs.python.org/3/license.html>
- [10] *Co je open source* [online]. [cit. 2021-5-6]. Dostupné z: <https://www.mioweb.cz/slovnicek/open-source/>
- [11] *Python: Pros and Cons* [online]. [cit. 2021-03-04]. Dostupné z: <https://scholarlyoa.com/python-pros-and-cons/>
- [12] *IDLE* [online]. [cit. 2021-5-6]. Dostupné z: <https://docs.python.org/3/library/idle.html>
- [13] *Python IDLE Reviews* [online]. [cit. 2021-5-6]. Dostupné z: <https://www.trustradius.com/products/python-idle/reviews?qs=pros-and-cons>
- [14] *The Python Logo* [online]. [cit. 2021-03-04]. Dostupné z: <https://www.python.org/community/logos/>

[15] PŠENČÍKOVÁ, Jana. *Algoritmizace*. Kralice na Hané: Computer Media, c2007. ISBN 80-86686-80-9.

[16] VYSTAVĚL, Radek. *Moderní programování: učebnice pro středně pokročilé*. Ondřejov: moderní Programování, 2008. ISBN 978-80-903951-2-1.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

RVP Rámcový vzdělávací program.

ŠVP Školní vzdělávací program.

IDLE Integrated Development and Learning Environment

IDE Vývojové prostředí

int Integer

sqrt Square root

bool Boolean

SEZNAM OBRÁZKŮ

Obrázek 1 – Přehled vyučovaných programovacích jazyků, zdroj: vlastní.....	13
Obrázek 2 – Logo jazyku Python, zdroj: [14]	15
Obrázek 3 – IDLE Shell 3.9.2, zdroj: vlastní.....	17
Obrázek 4 – Výukové prezentace, zdroj: vlastní	19

SEZNAM TABULEK

Tabulka 1 – Náplň vyučování na oboru Informační technologie, zdroj: [3]	10
Tabulka 2 – Časový harmonogram vyučovacích hodin.....	24
Tabulka 3 – Vyučovací hodina: Algoritmus - úvodní hodina	25
Tabulka 4 – Vyučovací hodina: Vývojové diagramy	27
Tabulka 5 – Vyučovací hodina: Algoritmus - zápis	30
Tabulka 6 – Vyučovací hodina: Proměnné, datové typy, vstupy, výstupy.....	33
Tabulka 7 – Vyučovací hodina: Cykly I.....	35
Tabulka 8 – Vyučovací hodina: Cykly II.....	37
Tabulka 9 – Vyučovací hodina: Cykly III.	40
Tabulka 10 – Vyučovací hodina: Procvičování tvorby složitějších algoritmů.....	43
Tabulka 11 – Vyučovací hodina: Třídící algoritmy – Bubble sort.....	46
Tabulka 12 – Vyučovací hodina: Algoritmizace - shrnutí.....	49
Tabulka 13 – Vyučovací hodina: Test – Algoritmizace, Úvod do programování.....	51
Tabulka 14 – Vyučovací hodina: Python - úvod	53
Tabulka 15 – Vyučovací hodina: Python – první programování.....	55
Tabulka 16 – Vyučovací hodina: Python – zápis proměnných, cyklů.....	58
Tabulka 17 – Vyučovací hodina: Python – tvorba jednoduchých programů	61
Tabulka 18 – Vyučovací hodina: Python – přepisování vývojových diagramů.....	63
Tabulka 19 – Vyučovací hodina: Python – tvorba komplexního programu I.	68
Tabulka 20 – Vyučovací hodina: Python – tvorba komplexního programu II.	71
Tabulka 21 – Vyučovací hodina: Test - Python	76

SEZNAM PŘÍLOH

Příloha P 1: Test algoritmizace – odpovědi

Příloha P 2: Test Python – odpovědi

Příloha P 3: Elektronická příloha na CD

PŘÍLOHA P 1: TEST ALGORITMIZACE - ODPOVĚDI

Algoritmizace – test

Varianta A.

Jméno:

Třída:

Datum:

V první části testu je 5 otázek po 1 bodu, dohromady tedy 5 bodů. Ve druhé části je zadán jeden algoritmus za maximální počet 10 bodů. Celkový počet bodů je tedy 15.

První část

1. Definujte pojem cyklus [1 bod]

Cykly slouží k vícenásobnému opakování stejné části programu a mají

.....

v programování široké využití.

.....

.....

2. Vypište 3 různé formy zápisu algoritmů [1 bod]

Přirozený jazyk

.....

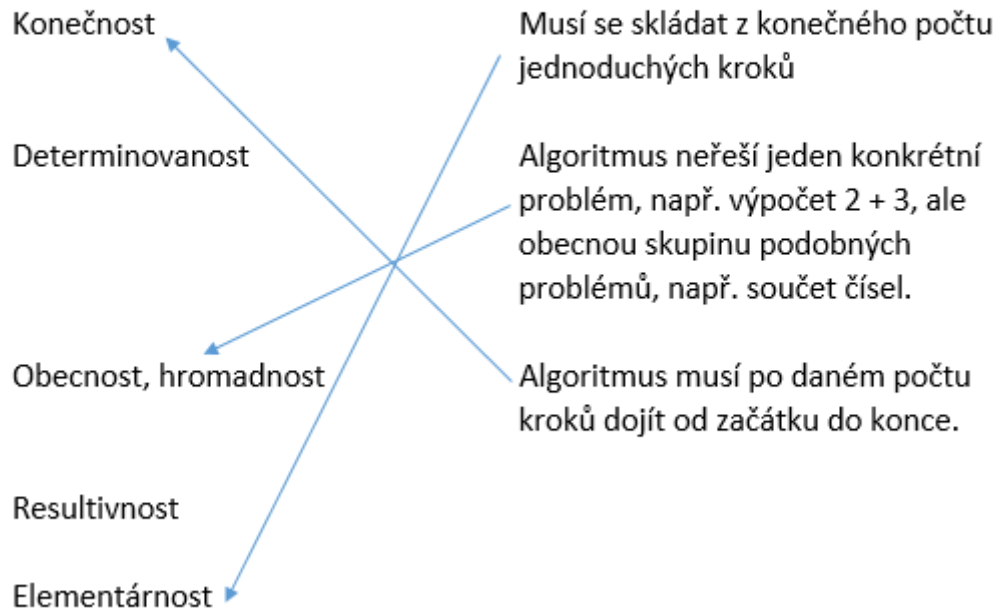
Programovací jazyk

.....

Grafická forma (vývojový diagram)

.....

3. Z možností uvedených ve sloupečcích utvořte dvojice, které k sobě patří (dvojice zřetelně spojte čarou), dvě možnosti v levém sloupečku jsou navíc, takže zůstanou nespojené [1 bod]



4. Přečtěte si následující text a odpovězte na otázku [1 bod]

Tento cyklus má stanovený pevný počet opakování a hlavně obsahuje tzv. řídicí proměnnou (celočíselnou), ve které se postupně během běhu cyklu mění hodnoty.

Který typ cyklu je popsán?

Cyklus for

.....

5. U každého z následujících výroku označte, zda je tento výrok pravdivý (napište +) nebo zda je tento výrok nepravdivý (napište -). [1 bod]

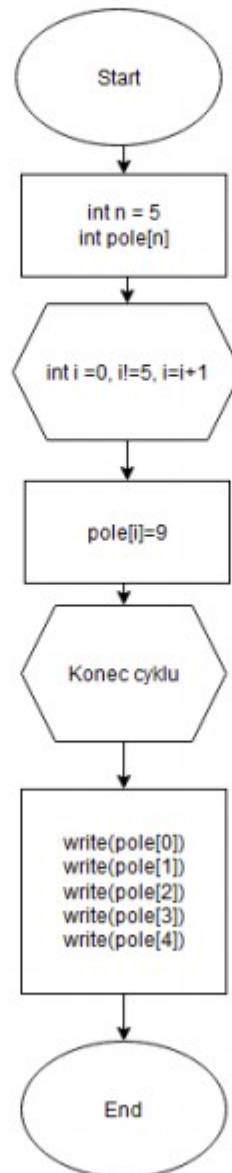
Operace modulo nám vypíše zbytek po celočíselném násobení. -

Datový typ Boolean může nabývat pouze dvou hodnot true:1/false:0. +

Nestabilní třídící algoritmus, nám při seřazení stejné vstupní množiny dá vždy stejný výsledek. -

Druhá část

Vytvořte algoritmus, zapsaný pomocí vývojového diagramu, kde na začátku vytvoříte pole s 5 pozicemi. Do všech těchto pozic vepíšete za pomoci cyklu for číslici 9. Poté všechny pozice pole ručně vypíšete. [10 bodů]



Algoritmizace – test

Varianta B.

Jméno:

Třída:

Datum:

V první části testu je 5 otázek po 1 bodu, dohromady tedy 5 bodů. Ve druhé části je zadán jeden algoritmus za maximální počet 10 bodů. Celkový počet bodů je tedy 15.

První část

1. Definujte pojem algoritmus[1 bod]

Algoritmus je posloupnost operací, při které v konečném počtu kroků

.....
se pro přípustná vstupní data, dojde ke správným výsledkům.
.....
.....

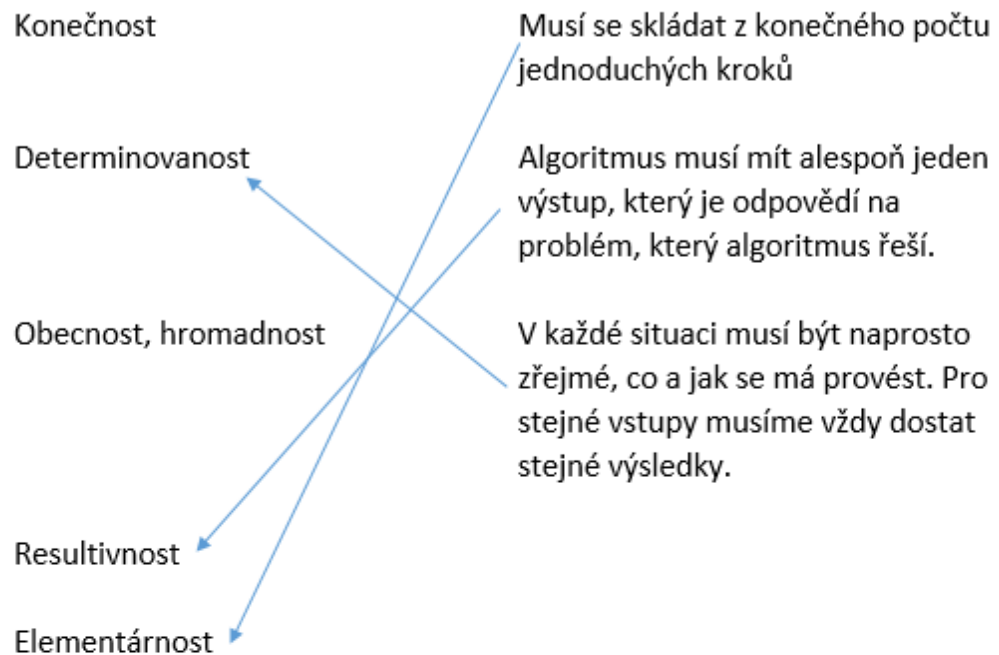
2. Vypište 3 různé druhy cyklů [1 bod]

for

.....
while

.....
do-while (repeat-until)
.....

3. Z možností uvedených ve sloupečcích utvořte dvojice, které k sobě patří (dvojice zřetelně spojte čarou), dvě možnosti v levém sloupečku jsou navíc, takže zůstanou nespojené [1 bod]



4. Přečtěte si následující text a odpovězte na otázku [1 bod]

Tento cyklus opakuje příkazy v bloku, dokud platí podmínka. Podmínka je uvedena na začátku cyklu, takže pokud by hned na začátku podmínka neplatila, cyklus se neprovede ani jednou.

Který typ cyklu je popsán?

Cyklus while

.....

5. U každého z následujících výroku označte, zda je tento výrok pravdivý (napište +) nebo zda je tento výrok nepravdivý (napište -). [1 bod]

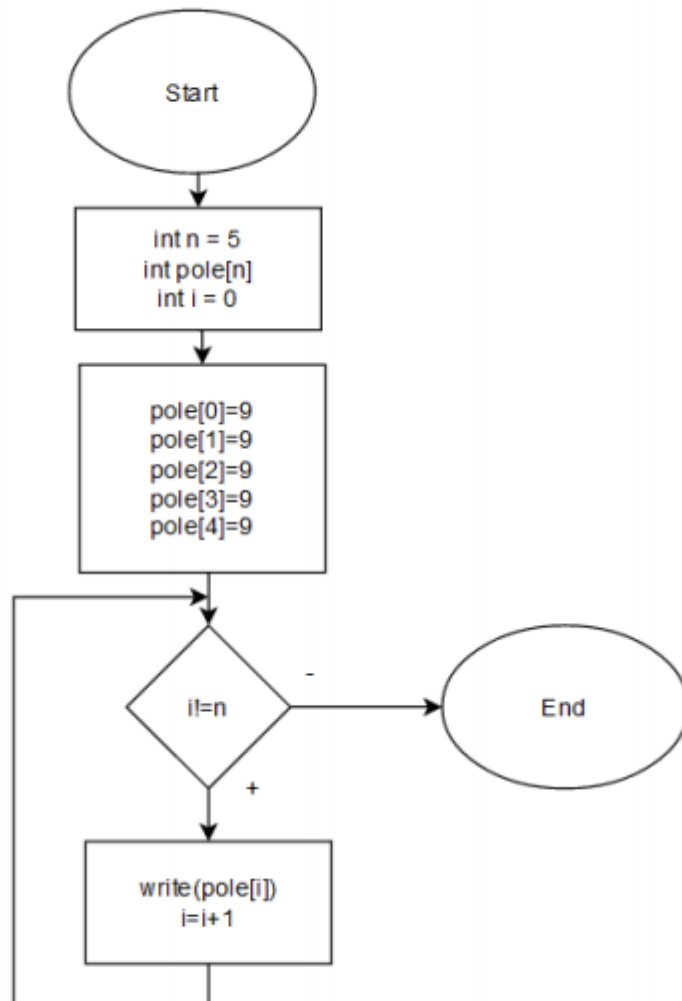
Operace $\text{sqrt}(a)$ nám vypočítá druhou mocninu z čísla a . -

Datový typ Integer může obsahovat pouze celočíselnou hodnotu. +

Všechny třídící algoritmy budou stejnou vstupní množinu třídit stejně dlouho. -

Druhá část

Vytvořte algoritmus, zapsaný pomocí vývojového diagramu, kde na začátku vytvoříte pole s 5 pozicemi. Do všech těchto pozic ručně zapíšete číslici 9. Poté všechny pozice s využitím cyklu while vypíšete. [10 bodů]



PŘÍLOHA P 2: TEST PYTHON – ODPOVĚDI

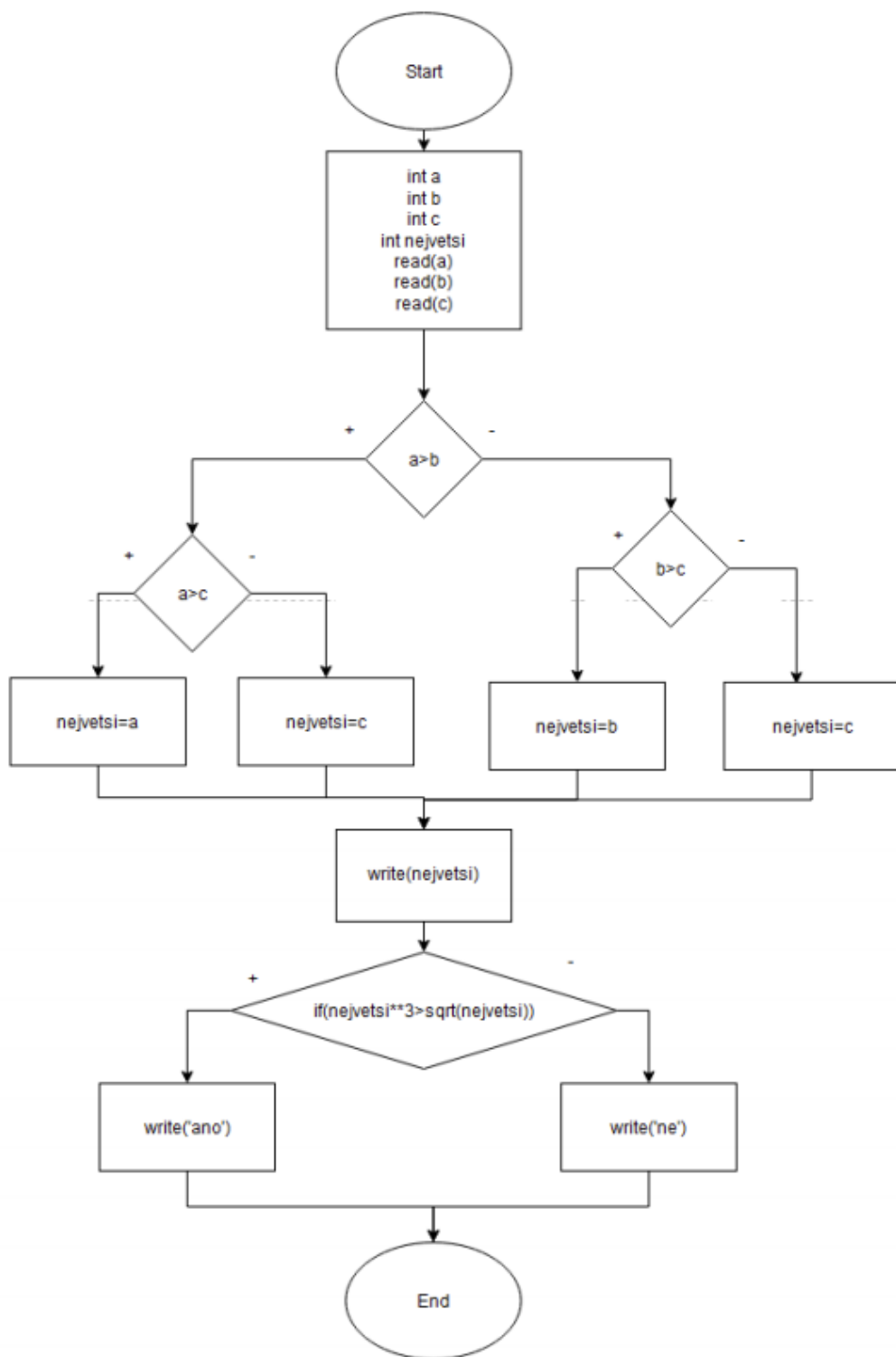
Algoritmizace – Python

Varianta A.

Zadání:

Vytvořte vývojový diagram a napište program, kde zadáme z konzole celočíselné hodnoty proměnných a,b,c. Potom program rozhodne, která proměnná z trojice je největší a vytiskne ji. Dále, pokud 3. mocnina největší proměnné je větší než druhá odmocnina z největší proměnné, vypíše ano, pokud je menší, vypíše ne.

Řešení:



```

import math

print('zadejte a')
a = int(input())
print('zadejte b')
b = int(input())
print('zadejte c')
c = int(input())

if(a>b):
    if(a>c):
        nejvetsi=a
    else:
        nejvetsi=c
else:
    if(b>c):
        nejvetsi=b
    else:
        nejvetsi=c

print(nejvetsi)

if(nejvetsi**3>math.sqrt(nejvetsi)):
    print('ano')
else:
    print('ne')

```

```

zadejte a
1
zadejte b
1
zadejte c
1
1
ne

```

```

zadejte a
2
zadejte b
5
zadejte c
6
6
ano

```

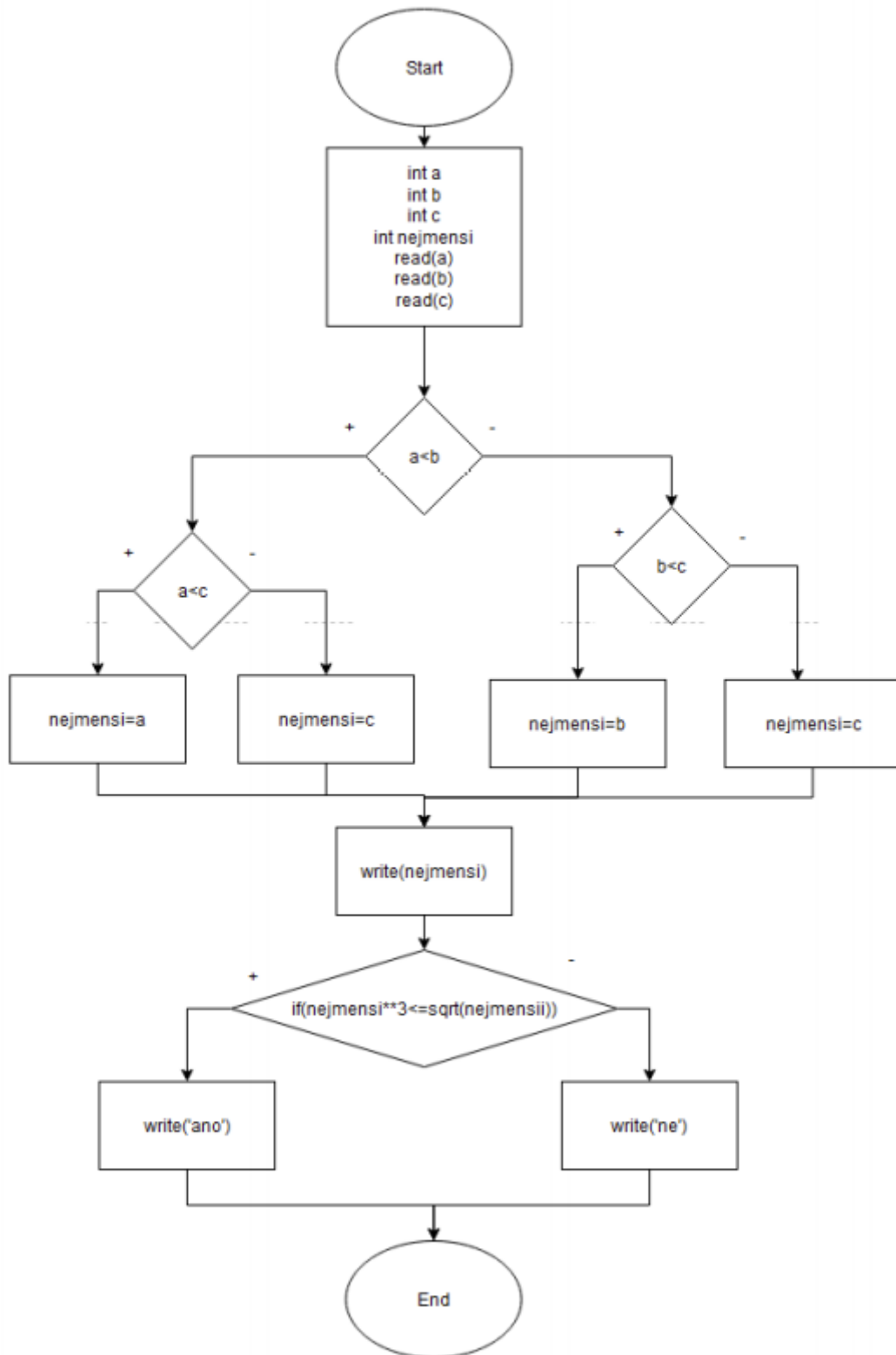
Algoritmizace – Python

Varianta B.

Zadání:

Vytvořte vývojový diagram a napište program, kde zadáme z konzole celočíselné hodnoty proměnných a, b, c . Potom program rozhodne, která proměnná z trojice je nejmenší a vytiskne ji. Dále, pokud 3. mocnina nejmenší proměnné je menší nebo rovna druhé odmocnině z nejmenší proměnné, vypíše ano, pokud je menší, vypíše ne.

Řešení:



```
import math

print('zadejte a')
a = int(input())
print('zadejte b')
b = int(input())
print('zadejte c')
c = int(input())

if(a<b):
    if(a<c):
        nejmensi=a
    else:
        nejmensi=c
else:
    if(b<c):
        nejmensi=b
    else:
        nejmensi=c

print(nejmensi)

if(nejmensi**3<=math.sqrt(nejmensi)):
    print('ano')
else:
    print('ne')
```

```
zadejte a
1
zadejte b
1
zadejte c
1
1
1
ano
```

```
zadejte a
3
zadejte b
2
zadejte c
5
2
ne
```


PŘÍLOHA P 3: ELEKTRONICKÁ PŘÍLOHA NA CD

Struktura přiloženého CD:

- dotaznik.docx
- idle_navod.docx
- Majer_DP.pdf
- rozvrzeni_hodin.docx
- test_algoritmizace.docx
- test_algoritmizace_odpovedi.docx
- test_python.docx
- test_python_odpovedi.docx
- prezentace
 - 1_algoritmus.pptx
 - 2_vyvojove_diagramy.pptx
 - 4_promenne_datove_typy.pptx
 - 5_cykly.pptx
 - 8_prace_s_poli.pptx
 - 9_tridici_algoritmy.pptx
 - 10_algoritmizace_shrnuti.pptx
 - 11_uvod_do_programovani.pptx
 - 12_python_uvod.pptx
 - 14_python_promenne_cykly.pptx