



Zadání bakalářské práce

Název:	Návrh a implementace platformy pro výpočet termodynamiky budov
Student:	Vojtěch Prendký
Vedoucí:	doc. Ing. Ivan Šimeček, Ph.D.
Studijní program:	Informatika
Obor / specializace:	Informační systémy a management
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	do konce letního semestru 2022/2023

Pokyny pro vypracování

Společnost Rekom, v.d. se zabývá zkoumáním a optimalizací termodynamiky budov. Pro iterační řešení výpočtů využívá platformu MS Excel. Spouštění jednotlivých iterací se provádí ručně, současné řešení je proto velmi časově náročné. Cílem práce je vytvořit studii proveditelnosti nasazení efektivnějšího řešení.

- 1) Analyzujte požadované vstupní a výstupní formáty a výpočetní část současného řešení.
- 2) Identifikujte slabá místa současného řešení.
- 3) Diskutujte vhodné platformy pro implementaci.
- 4) Proveďte analýzu nákladů a výnosů spojených s vývojem a nasazením navrhovaného řešení.



**FAKULTA
INFORMAČNÍCH
TECHNOLGIÍ
ČVUT V PRAZE**

Bakalářská práce

Návrh a implementace platformy pro výpočet termodynamiky budov

Vojtěch Prendký

Katedra softwarového inženýrství

Vedoucí práce: doc. Ing. Ivan Šimeček, Ph.D.

10. května 2022

Poděkování

Rád bych poděkoval mému vedoucímu doc. Ing. Ivanu Šimečkovi, Ph.D. za cenné postřehy při tvorbě bakalářské práce. Panu Ing. Janu Blažíčkovi ze společnosti Rekom, v. d. za konzultace a data k problému. Mému příteli Ing. Miroslavovi Brzobohatému za zkušenosti z jeho tvorby závěrečných psaní. Mé rodině za podporu v průběhu celého studia.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 2373 odst. 2 zákona č. 89/2012 Sb., občanský zákoník, ve znění pozdějších předpisů, tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu) licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 10. května 2022

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2022 Vojtěch Prendký. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí a nad rámec oprávnění uvedených v Prohlášení na předchozí straně, je nezbytný souhlas autora.

Odkaz na tuto práci

Prendký, Vojtěch. *Návrh a implementace platformy pro výpočet termodynamiky budov*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2022.

Abstrakt

Výstupem práce je studie proveditelnosti zefektivnění výpočtu přenosu tepla v budovách společnosti Rekom, v. d. Konkrétně návrh opatření na optimalizaci stávajícího řešení v MS Excel či návrh nového řešení ve zvolené platformě. Součástí práce je ekonomicko-provozní zhodnocení současného a navrhovaného řešení. Práce se zpočátku zabývá přiblížením problematiky výpočtu distribuce tepla v budovách a zhodnocením stávající implementace. Na základě získaných informací vylučuje možnost rozvoje stávajícího řešení. Poté vybírá skupinu programovacích jazyků vhodných pro novou implementaci výpočtu, které porovnává s ohledem na rychlost a jejich možnosti. Jako nejvýhodnější byla zvolena implementace nového řešení v jazyce C++. Z analýzy nákladů a výnosů bylo zjištěno, že investice do nového systému se vyplatí již během prvního roku po nasazení. Výsledek této práce umožňuje společnosti Rekom, v. d. zvýšit efektivitu procesů ve společnosti bez rizika zmařené investice.

Klíčová slova studie proveditelnosti, návrh řešení, tepelný přenos budov, metoda ustálených toků, Rekom, v. d., efektivita programu, Microsoft Excel, C++

Abstract

The result of the thesis is a feasibility study of the efficiency of heat transfer calculation in the buildings of Rekom, v. d. Specifically, the proposal of measures to optimize the existing solution in MS Excel or the proposal of a new solution in the chosen platform. The work includes an economic and operational evaluation of the current and proposed solution. The thesis initially deals with an introduction to the problem of heat distribution calculation in buildings and an evaluation of the current implementation. Based on the information obtained, it excludes the possibility of developing the existing solution. It then selects a group of programming languages suitable for the new implementation of the calculation, comparing them with respect to speed and their capabilities. The implementation of the new solution in C++ was chosen as the most advantageous. From the cost-benefit analysis it was found that the investment in the new system pays off within the first year after deployment. The result of this work allows Rekom, v. d. to increase the efficiency of the company's processes without the risk of a wasted investment.

Keywords feasibility study, solution design, heat transfer of buildings, steady flow method, Rekom, v. d., program efficiency, Microsoft Excel, C++

Obsah

Úvod	1
1 Distribuce tepla v budovách	3
1.1 Stručné seznámení	3
1.2 Výpočet ztrát metodou ustálených tepelných toků	4
1.2.1 Přenos tepla	5
1.2.1.1 Výpočet návrhové tepelné ztráty prostupem	5
1.2.1.2 Výpočet tepelné ztráty větráním	7
1.2.1.3 Výpočet tepelného zisku vlivem sálání	8
2 Rozbor stávajícího řešení	9
2.1 O společnosti	9
2.2 Technologické zázemí	10
2.2.1 Platforma výpočtu	10
2.2.2 Visual Basic for Applications	10
2.2.3 Stroje	10
2.3 Stávající řešení	10
2.3.1 Vstupní a výstupní soubory	11
2.3.1.1 Formát vstupu	11
2.3.1.2 Formát výstupu	11
2.3.2 Výpočet	11
2.3.2.1 Složitost algoritmu	11
2.3.3 Limity současného řešení	12
2.3.4 Překážky v rozvoji stávajícího řešení	13

2.3.5	Ekonomická zátěž současného stavu	13
3	Systémy vhodné pro novou implementaci	15
3.1	Vhodné systémy	15
3.2	Porovnání rychlosti jazyků	15
3.3	Kompilované a interpretované jazyky	17
3.4	Výsledek	18
4	Studie proveditelnosti	19
4.1	Přehled a cíle projektu	19
4.2	Analýza a návrh řešení	19
4.2.1	Analýza požadavků	19
4.2.1.1	Funkční požadavky	19
4.2.1.2	Nefunkční požadavky	20
4.2.2	FURPS analýza	20
4.2.3	Návrh řešení a naplnění požadavků	21
4.2.3.1	Zvolená technologie	21
4.2.3.2	Návrh řešení	21
4.2.3.3	Naplnění požadavků	22
4.3	Vývoj a nasazení systému	22
4.3.1	Zdroje	22
4.3.2	Časový plán	22
4.3.3	Finanční a ekonomická analýza	23
4.3.3.1	Náklady	23
4.3.3.2	Výnosy	24
4.4	Zhodnocení rizik a příležitostí	24
4.4.1	Rizika	24
4.4.2	Příležitosti	24
4.5	Shrnutí výsledků	25
5	Aktuální stav a možnosti dalšího rozvoje	27
5.1	Aktuální stav	27
5.2	Možnosti dalšího vývoje	27
	Závěr	29
	Literatura	31

A	Seznam použitých zkratek	35
B	Obsah přiloženého CD	37

Seznam obrázků

1.1	Tepelné toky v místnosti	4
1.2	Způsoby přenosu tepla	5
1.3	Prostup tepla konstrukcí	6
1.4	Ztráta tepla větráním	7
1.5	Zisk tepla vlivem sálání	8

Seznam tabulek

3.1	Programovací jazyky dle typu	16
3.2	Použité testy a jejich popis	16
3.3	Porovnání rychlosti programovacích jazyků pro binární stromy . . .	17
3.4	Normalizované souhrnné porovnání rychlosti programovacích jazyků	17
4.1	Pokrytí FURPS požadavky	21
4.2	Tabulka nákladů	23
4.3	Tabulka úspor	24
4.4	Souhrn nákladů a výnosů	25

Úvod

Každá společnost řeší dilema, jestli se vyplatí investice do systému, který slibuje zvýšit efektivitu některých z procesů ve společnosti. Stejně dilema řeší společnost Rekom, v.d., která se zabývá zkoumáním a výpočty termodynamické účinnosti budov. Jako platforma pro výpočty byl vybrán tabulkový procesor Microsoft Excel, tato volba však při současném objemu zakázek společnosti představuje poměrně velkou zátěž. Výpočty jsou pomalé, vyžadují lidské vstupy a často je nutné projekty uměle dělit na menší části. Společnost by tak ráda výpočty optimalizovala, či převedla do vhodnějšího systému.

Pro odpovědné rozhodnutí potřebuje společnost zjistit, jaké kroky budou mít na zrychlení výpočtu jaký vliv a jak náročné bude je zavést do praxe. Tato práce se proto zabývá zkoumáním možných metod zvýšení efektivity zmíněných výpočtů a systémy umožňujícími efektivní výpočty s velkým množstvím vstupních dat. Dále zkoumá současný stav společnosti, finanční náklady a personální nároky spojené s užíváním aktuální verze systému a současný systém výpočtů. V úvodu praktické části navrhuje vhodné řešení pro optimalizaci procesů společnosti. Hlavním obsahem praktické části je studie proveditelnosti nasazení navrhovaného řešení.

Cílem práce je seznámit čtenáře s problematikou distribuce tepla v budovách, prozkoumat současné systémy umožňující efektivně provádět rozsáhlé výpočty a zhodnotit současné řešení. Na základě získaných informací pak zvolit vhodný způsob zefektivnění výpočtů ve společnosti Rekom, v. d. a zpracovat studii proveditelnosti nasazení navrhovaného řešení.

Distribuce tepla v budovách

1.1 Stručné seznámení

Zkoumání distribuce tepla v budovách v posledních letech nabývá na důležitosti. Ve velkých budovách je poměrně náročné sledovat tepelné úniky a mosty. Ne všechny místnosti, zvláště pak ve velkých komplexech, mají stejnou požadovanou teplotu. Teplotní požadavky v kancelářích či učebnách budou zcela jiné než například v podzemních garážích. Ve vytápěných místnostech se teplota určí a následně se pomocí otopné soustavy udržuje. V nevytápěných místnostech však takovou možnost nemáme, teplota se v nich určuje odhadem, případně užitím nějaké konvenční hodnoty.

Místnosti v budovách jsou od sebe odděleny stavebními prvky jako jsou například stěny, podlahy, dveře a podobně. Tyto stavební prvky však mají různé izolační vlastnosti, teploty v místnostech se tak navzájem ovlivňují. Často se pak stane, že některé místnosti jsou neustále přetápěny, načež dochází k neúměrným ztrátám tepla větráním. Jiné místnosti je naopak nemožné spolehlivě vytopit na požadovanou teplotu, což přináší značný diskomfort uživatelům takové místnosti.

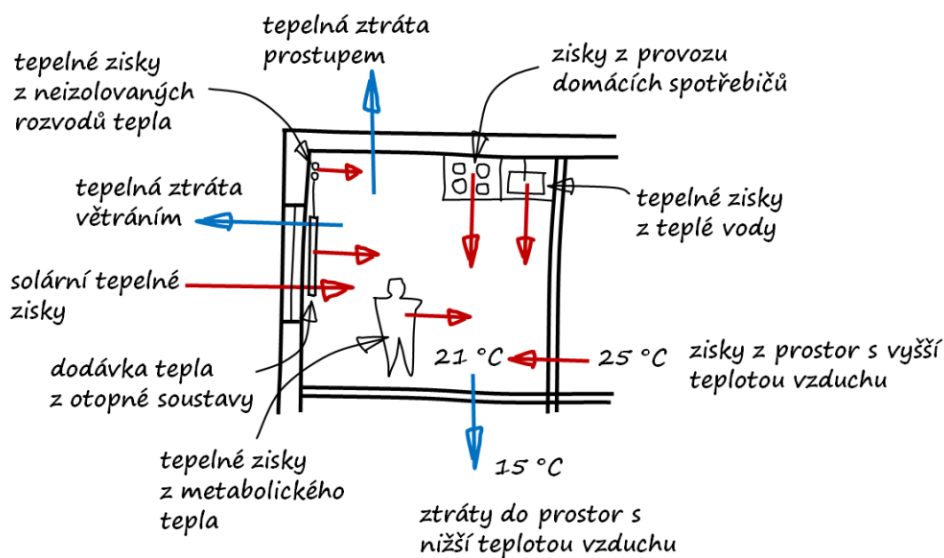
Nové budovy se navrhují jako komplexní systémy s propracovanou termodynamikou, často do detailu vymodelovanou v průběhu návrhu budovy. Návrh budovy je často podřízen optimálním tepelným vlastnostem, zejména použité materiály jsou voleny podle jejich tepelných vlastností. Správně navržené budovy přináší uživatelům vyšší komfort, ale také nižší náklady na vytápění. Při úpravách stávajících budov ale takové možnosti stavitelé často nemají. Je proto třeba nejprve pečlivě prozkoumat stávající situaci. S rozvojem mo-

derních diagnostických nástrojů je možné provádět čím dál detailnější měření. Tím se ale také zvyšují požadavky na rozsah prováděných výpočtů. S jejich pomocí je možné zjistit skutečnou potřebu tepla v jednotlivých místnostech, což umožní správně seřadit otopnou soustavu a předvídat vliv různých provozních a stavebních opatření na komfort a provozní náklady. [2], [8], [24]

1.2 Výpočet ztrát metodou ustálených tepelných toků

Řešení společnosti Rekom, v. d. počítá vnitřní teploty všech místností, včetně těch nevytápěných, metodou ustálených tepelných toků. Do konečné bilance je zahrnut i vliv větrání a solárních zisků. Jedinou neznámou v celkové tepelné bilanci tak jsou vnitřní zisky, například metabolické teplo nebo teplo vytvořené domácími spotřebiči. Jejich efekt je ovšem již promítnut do náměrů poměrových indikátorů, který je součástí vstupu programu, proto je není třeba zvlášť zohledňovat. Pro názornost je přiložen obrázek 1.1, [18], [24]

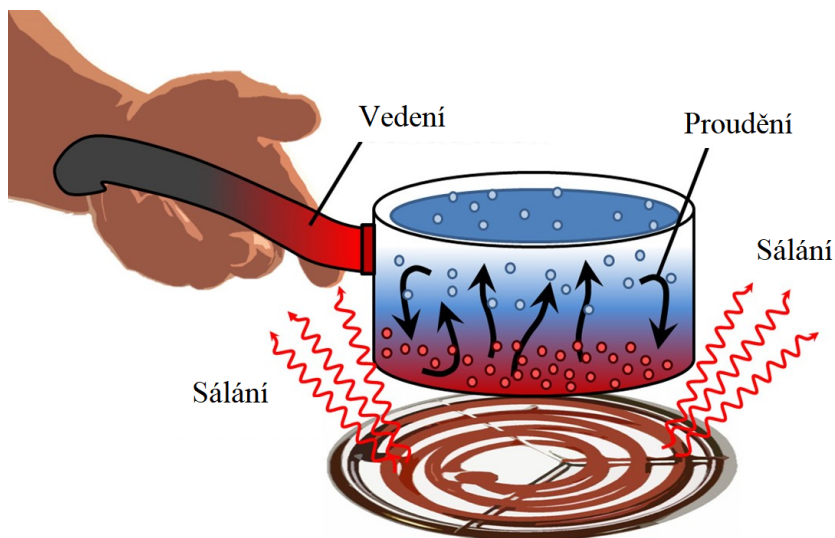
Obrázek 1.1: Tepelné toky v místnosti



[19]

1.2.1 Přenos tepla

Přenos tepla je fyzikální jev výměny tepelné energie mezi dvěma systémy. Z míst, která mají vyšší teplotu, se energie přenáší do oblastí s nižší teplotou. Přenos tepla 1.2 je možné rozdělit do tří kategorií: vedení (kondukce), proudění (konvekce) a sálání (záření, radiace). Pokud pomíneme proměnlivou venkovní teplotu a intenzitu slunečního svitu, lze vedení a sálání tepla v budovách poměrně přesně změřit. Proudění se v praxi určuje pouze jako odhad ztráty větráním. [2], [22]



Obrázek 1.2: Způsoby přenosu tepla

[22]

1.2.1.1 Výpočet návrhové tepelné ztráty prostupem

Nejdůležitější částí výpočtu je určení tepelných ztrát vzniklých prostupem tepla plochami, které oddělují daný prostor od okolí. Ztráty jsou znázorněny na obrázku 1.3. Vypočítáme je z rozdílu teplot před a za jednotlivými konstrukcemi a z měrného tepelného toku, který je přímo úměrný výměře a tepelně technickým vlastnostem každé z konstrukcí. Z uvedeného vzorce 1.1 je patrné,

1. DISTRIBUCE TEPLA V BUDOVÁCH

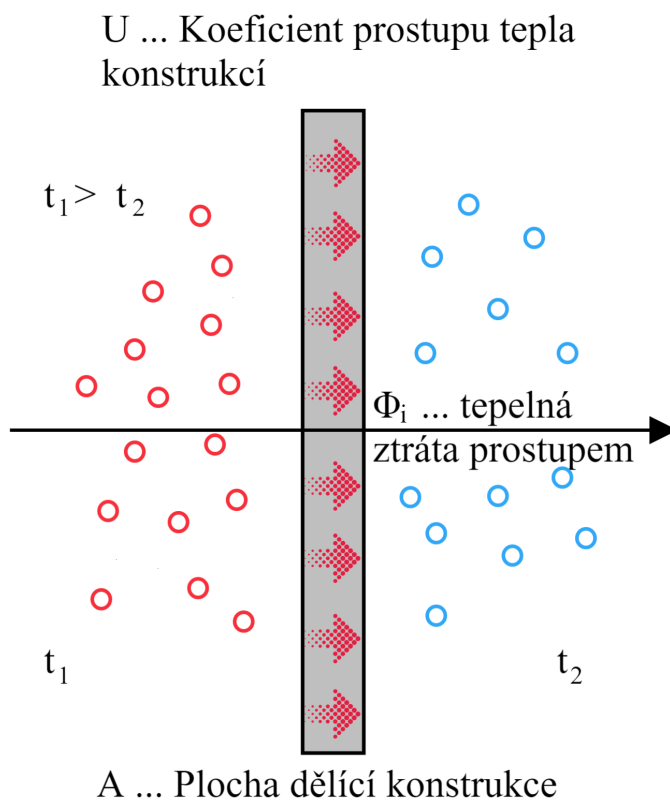
že o množství prostupujícího tepla rozhoduje stejnou měrou jak rozdíl teplot Δt , tak kvalita dělicí konstrukce vyjádřená měrným tepelným tokem $U \cdot A$. [3], [17], [20], [22], [23]

$$\Phi_i = H \cdot \Delta t_i = U \cdot A \cdot \Delta t_i \quad (1.1)$$

kde:

- A plocha konstrukce [m^2]
- H součinitel tepelné ztráty [W/K]
- Δt_i teplotní rozdíl [$^{\circ}C$]
- U součinitel prostupu tepla [$W/m^2 \cdot K$]
- Φ_i návrhová tepelná ztráta [W]

Obrázek 1.3: Prostup tepla konstrukcí



[17]

1.2.1.2 Výpočet tepelné ztráty větráním

Nedílnou součástí tepelné bilance jsou ztráty vzniklé větráním, jejichž znázornění můžete vidět na obrázku 1.4. Parametry vzorce 1.2 nepotřebují blíže představovat, kromě parametru n , který představuje intenzitu větrání. Hygieniky doporučená hodnota je polovina objemu vzduchu v místnosti za hodinu, experti na energetiku doporučují třetinu objemu za hodinu. Obvykle se užívá hodnota 0,4, výpočet ale zanedbává tloušťku stěn, reálná hodnota parametru se tedy pohybuje v rozmezí $0,4 \leq x \leq 0,5$. [3], [20], [22]

$$Q_v = n \cdot V \cdot \sigma \cdot c \cdot \Delta t_i \quad (1.2)$$

kde:

c měrná tepelná kapacita vzduchu [$J/kg \cdot K$]

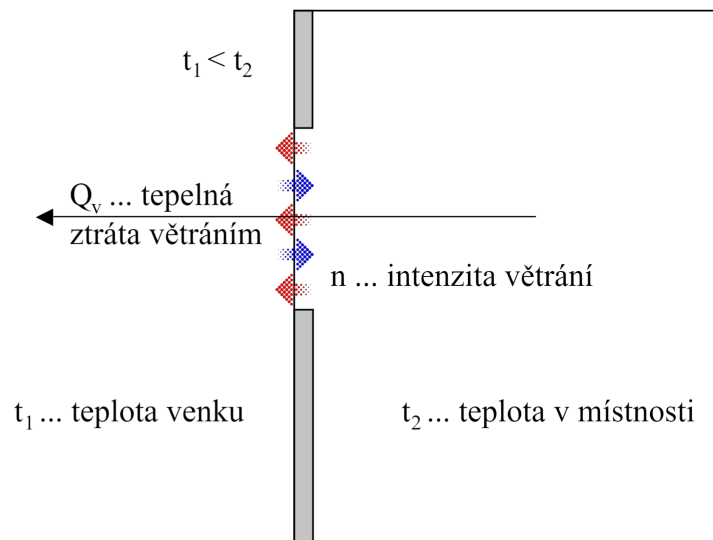
n intenzita větrání [hod^{-1}]

Δt_i teplotní rozdíl [$^{\circ}C$]

V objem místnosti [m^3]

σ objemová hmotnost vzduchu [kg/m^3]

Obrázek 1.4: Ztráta tepla větráním



1.2.1.3 Výpočet tepelného zisku vlivem sálání

Poslední část výpočtu určuje množství tepla získaného ze slunečního záření, viz obrázek 1.5. Všechny parametry rovnice 1.3 jsou programu poskytnuty ve vstupu, nebo se jedná o konstanty. [3], [21], [22]

$$Q_s = \epsilon \cdot c_0 \cdot \Psi \cdot A \cdot [(T_{ip}/100)^4 - (T_v/100)^4] \quad (1.3)$$

kde:

A plocha konstrukce [m^2]

c_0 konstanta sálání dokonale černého tělesa $c_0 = 5,775 [W \cdot m^2 \cdot K^{-4}]$

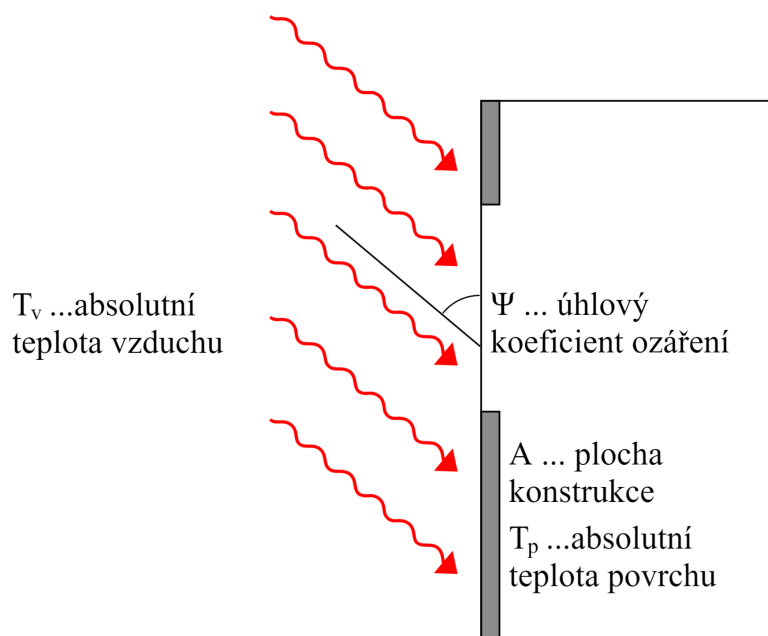
T_{ip} absolutní teploty povrchů [K]

T_v absolutní teplota vzduchu [K]

ϵ emisní součinitel [1]

Ψ úhlový koeficient ozáření [1]

Obrázek 1.5: Zisk tepla vlivem sálání



Rozbor stávajícího řešení

2.1 O společnosti

Společnost Rekom, v. d. vyvinula propracovaný systém měření a výpočtů, který slibuje s velkou přesností určovat slabé články v otopném systému budovy. Cílem zkoumání je:

- Nastavení efektivního ústředního vytápění, čímž se zajistí nižší provozní náklady v důsledku seřízení výkonové regulace otopné soustavy, maximálního využívání bezplatných tepelných zisků pro účel vytápění na úkor tepla, které je draze nakupováno či vyráběno vlastními zdroji
- Výpočet objektivní energetické náročnosti vytápěných prostor. To je nezbytná podmínka spravedlivého rozúčtování tepelných nákladů v duchu hesla „stejně velké prostory vytápěné na stejnou teplotní úroveň musí mít rovněž stejné náklady na teplo“, zohlednění všech relevantních vlivů, včetně slunečního záření.
- Objektivní určení spotřeby tepla, které zahrnuje přesné vyčíslení všech složek tepelné bilance včetně tepelných zisků, reálný obrázek o využívání energie z otopné soustavy, zohlednění vlivu přestupů tepla mezi byty a do společných prostor, eliminace tepelného parazitování.
- Přehledné rozúčtování včetně analýzy stavu v podobě on-line domovního konta. To poskytne průběžný přehled o fungování tepelného hospodaření v objektu. Zároveň slouží jako analytický podklad pro řešení problémů v oblasti vytápění a rozúčtování tepla.

Tato práce se zabývá jednou z důležitých částí hodnocení otopného systému budov. Tou je výpočet ztrát metodou ustálených tepelných toků. V něm se zjišťují tepelné toky mezi vytápěnými a nevytápěnými prostory a upravují se parametry vytápění tak, aby se docílilo téměř nulové bilance toku energií. [1], [2]

2.2 Technologické zázemí

2.2.1 Platforma výpočtu

Současná verze výpočtů je implementovaná v programu Microsoft Excel, což je tabulkový procesor vyvíjený společností Microsoft corporation. Program MS Excel komplexní nástroj umožňující snadno vytvářet tabulky nebo celé sešity s více tabulkami. V tabulkách lze s pomocí zabudovaných funkcí vytvářet a vyhodnocovat rovnice. Program ale umožňuje také vytváření jednoduchých automatizací s pomocí maker v jazyce Visual Basic for Applications.

2.2.2 Visual Basic for Applications

Visual Basic for Applications je jazyk vytvořený společností Microsoft corporation pro použití v programech balíku Microsoft Office. V Excelu umožňuje jazyk VBA uživatelům dále upravovat výpočty v tabulkách vytvářením maker, čímž umožňuje vytvářet jednoduché automatizace i lidem, kteří nejsou programátoři. Výpočty vytvořené s pomocí VBA jsou poměrně efektivní, přesto jsou až dvakrát pomalejší než srovnatelná implementace v C++. [10], [11], [12], [13]

2.2.3 Stroje

Pro výpočet se používají středně výkonné osobní počítače a laptopy. Pro jednoduchost uvažujme laptop vybavený procesorem Intel Core i5-8250U a 8 GB RAM. Počítače využívají systém Microsoft Windows 10.

2.3 Stávající řešení

Protože způsob implementace výpočtu je unikátní, představíme si výpočet pouze obecně.

2.3.1 Vstupní a výstupní soubory

2.3.1.1 Formát vstupu

Vstupní soubor stávajícího řešení je sešit programu MS Excel obsahující několik jednoduchých tabulek s daty získanými z ručního měření. První tabulka obsahuje údaje o místnostech, každý její řádek odpovídá jedné místnosti. Další tabulka obsahuje parametry dělicích příček. Poslední tabulka obsahuje okrajové podmínky jako je například nadmořská výška. Data z tabulek se manuálně importují do příslušných míst v sešitu výpočtu. Některé hodnoty se dokonce musí přidávat až v průběhu výpočtu. Konzistenci značení zajišťuje výhradně osoba provádějící výpočet, program je schopný detekovat pouze významné chyby v datech a není vždy schopný určit výskyt chyby.

2.3.1.2 Formát výstupu

Výstup výpočtu je realizovaný výhradně výpisem dat do tabulek sešitu řešení. Dílčí výsledky je nutné kopírovat v průběhu výpočtu mezi jednotlivými iteracemi. Konečné výsledky se pak kopírují či exportují do nové tabulky.

2.3.2 Výpočet

Současná verze výpočtu v několika cyklech střídavě zvyšuje cílovou teplotu v nevytápěných prostorách o teplo přijaté z okolních konstrukcí a ze slunečního záření. Poté se směr výpočtu obrátí a cílová teplota se snižuje o teplo vyzářené a ztracené větráním. Cykly se takto opakují, dokud se nedosáhne přibližně ustáleného stavu. V současnosti se výpočet přeruší při kolísání teplot $\leq 0,1^\circ\text{C}$, větší přesnost nemá z hlediska praktického využití význam. [2]

2.3.2.1 Složitost algoritmu

Abychom mohli určit výpočetní náročnost výpočtu a porovnat jej se současnou implementací, je nutné si nejdříve definovat pojem třída složitosti algoritmu.

Definice: Nechtě $f, g : \mathbb{N} \rightarrow \mathbb{R}$ jsou dvě funkce. Řekneme, že funkce $f(n)$ je třídy $O(g(n))$, jestliže existuje taková kladná reálná konstanta c , že pro skoro všechna n platí $f(n) \leq cg(n)$. Skoro všemi n se myslí, že nerovnost může selhat pro konečně mnoho výjimek, tedy že existuje nějaké přirozené n_0 takové, že

nerovnost platí pro všechna $n \geq n_0$. Funkci $g(n)$ se pak říká asymptotický horní odhad funkce $f(n)$.

Jinými slovy, dostatečně velký násobek funkce $g(n)$ shora omezuje funkci $f(n)$. Konečně mnoho výjimek se hodí tehdy, když má funkce $g(n)$ několik počátečních funkčních hodnot nulových či dokonce záporných, takže je nemůžeme „přebít“ jakkoliv vysokou konstantou c . [5]

V nejhorším případě musí výpočet u každé místnosti zohlednit všechny ostatní místnosti. Zvlášť se počítá pro každý typ příčky, například pokud místnosti oddělují dveře a stěna, jedná se o dva dílčí výpočty. Dělicích konstrukcí každé místnosti je ale konečně mnoho, lze je proto v odhadu složitosti zanedbat. Pokud máme n místností, v každém kroku iterace je i -tá místnost ovlivněna $n-i-1$ místnostmi. Celkový počet operací tedy činí až $\sum_{n=0}^{n-2} n-i-1 = \frac{n \cdot (n-1)}{2}$. Asymptotický horní odhad třídy složitosti je tedy $O(\frac{n \cdot (n-1)}{2}) = O(n^2 - n) = O(n^2)$.

2.3.3 Limity současného řešení

Kvůli uložení dat v tabulkách není možné data přímo adresovat. V každém kroku iterace se tak prohledává seznam místností a dělicích příček, aby se mohly započítat sousední místnosti. K prohledávání seznamů je použita funkce VLOOKUP, která ve výpočtu funguje tak, že postupně prohledává buňky v určeném sloupci, dokud nenalezne přesnou shodu. Protože data nejsou vždy seřazená, není možné spoléhat na indexaci, navíc v dokumentaci není přesně popsáno zrychlení umožněné indexováním dat. Takže funkce musí při každém průchodu projít seznam všech n místností. V každém kroku výpočtu se tedy provádí nejvýše n porovnání. Celý program tak náleží do třídy složitosti $O(n * n^2) = O(n^3)$. Současná podoba výpočtu a způsob přiřazování konstrukcí neumožňuje rozlišovat více než 80 druhů konstrukcí. Navíc kvůli časové a paměťové náročnosti současné implementace není možné zpracovávat projekty s více než 500 místnostmi, větší projekty se tak musí pracně rozdělovat a dopočítávat ručně. [9]

2.3.4 Překážky v rozvoji stávajícího řešení

Komplexní podoba výpočtů neumožňuje jednoduché odstranění všech neduhů. Změna označování konstrukcí by vynutila úpravu velké části programu. Uložení údajů v tabulkách také neumožňuje jejich přímou adresaci, která je nezbytná k odstranění opakovaného a velmi pomalého prohledávání dat. Je rovněž obtížné omezit spotřebu paměti, neboť v mnoha případech je nutné údaje mezi tabulkami kopírovat.

2.3.5 Ekonomická zátěž současného stavu

Délka výpočtu na dostupných počítačích je dle odhadu pana Ing. Blažička, zástupce společnosti Rekom, v. d., přibližně 200 minut pro průměrně velké projekty. Výpočty nejsou plně automatizované. Je proto nutné, aby po celou dobu výpočtu byl přítomný pracovník, který kontroluje výstupní hodnoty a spouští jednotlivé kroky výpočtu. Hodnotu času svých zaměstnanců firma odhaduje na 700 Kč / MH.

Kromě samotných nákladů je také nutné vzít v úvahu fakt, že časová a personální náročnost současné implementace je potenciální překážkou pro další rozvoj společnosti. V současnosti sice není nutné odmítat projekty kvůli vytíženosti, počet a rozsah zakázek společnosti se však průběžně zvyšuje.

Systemy vhodné pro novou implementaci

3.1 Vhodné systémy

Protože výpočet, kterým se práce zabývá, sestává především z opakovaného násobení, není třeba hledat platformu se zvláštními funkcemi. Pro realizaci výpočtu bude vhodný jakýkoliv programovací jazyk, který je dobře optimalizovaný v základních matematických operacích jako je sčítání, násobení a dělení. Zároveň musí umožňovat efektivní přístup k datům. Protože zmíněné vlastnosti má mnoho moderních programovacích jazyků, je vhodné porovnat jejich efektivitu.

3.2 Porovnání rychlosti jazyků

Tato práce si neklade za cíl zkoumat rychlost jednotlivých jazyků. Proto pouze přebírá výsledky ze studie srovnání rychlosti, využití paměti a efektivity programovacích jazyků, ve které autoři porovnávali 27 různých programovacích jazyků v různých algoritmičeských problémech. Podrobnosti jsou k nalezení v tabulkách 3.1 a 3.2. Výsledky jsou k nahlédnutí v tabulkách výsledků binárních stromů 3.3 a normalizovaných souhrnných výsledků 3.4. Platí, že nižší výsledek je lepší. [6]

3. SYSTÉMY VHODNÉ PRO NOVOU IMPLEMENTACI

Tabulka 3.1: Programovací jazyky dle typu

Typ	Programovací jazyk
Funkcionální	Erlang, F#, Haskell, Lisp, Ocaml, Perl, Racket, Ruby, Rust;
Imperativní	Ada, C, C++, F#, Fortran, Go, Ocaml, Pascal, Rust;
Objektově orientovaný	Ada, C++, C#, Chapel, Dart, F#, Java, JavaScript, Ocaml, Perl, PHP, Python, Racket, Rust, Smalltalk, Swift, TypeScript;
Skriptovací	Dart, Hack, JavaScript, JRuby, Lua, Perl, PHP, Python, Ruby, TypeScript;

[6]

Tabulka 3.2: Použité testy a jejich popis

Test	Popis
n-body	n-body simulace s dvojitou přesností
fannkuch - redux	Indexovaný přístup k malým celočíselným posloupnostem
spectral - norm	Vlastní čísla pomocí power metody
mandelbrot	Vytvoření přenositelné bitmapy Mandelbrotovy množiny
pidigits	Přenos aritmetiky s neomezenou přesností
regex - redux	Párování osmic DNA a nahrazení magic patterns
fasta	Generování a zápis náhodných sekvencí DNA
k - nucleotide	Aktualizace hashovací tabulky a řetězce k - nukleotidů
reverse - complement	Čtení náhodných sekvencí DNA a zápis jejich reverzních doplňků
binary - trees	Alokování, procházení a dealokace mnoha binárních stromů
chameneos - redux	Rendezvous symetrických vláken
meteor - contest	Hledání možných tvarů packing puzzle
thread - ring	Střídání vláken s předáváním jednoho tokenu

[6]

3.3. Kompilované a interpretované jazyky

Tabulka 3.3: Porovnání rychlosti programovacích jazyků pro binární stromy

Jazyk	Čas běhu (ms)	Využitá paměť (Mb)
C	1125,00	131,00
C++	1129,00	132,00
Rust	1263,00	180,00
Fortran	2112,00	133,00
Ada	2822,00	197,00
Java	3306,00	1120,00

[6]

Tabulka 3.4: Normalizované souhrnné porovnání rychlosti programovacích jazyků

Jazyk	Čas běhu oproti nejrychlejšímu	Využitá paměť oproti nejlepšímu
C	100%	117%
Rust	104%	154%
C++	156%	134%
Ada	185%	147%
Java	189%	601%
Chapel	214%	400%
Go	283%	105%
Pascal	302%	100%
Ocaml	309%	282%
C#	314%	285%

[6]

3.3 Kompilované a interpretované jazyky

Programovací jazyky se řadí do dvou hlavních skupin, podle způsobu jakým probíhá překlad kódu na strojové instrukce.

První skupinou jsou kompilované jazyky. Zdrojový kód je při kompilaci převeden přímo na strojové instrukce. Díky tomu jsou programy vytvořené v kompilovaných jazycích obvykle rychlejší a efektivnější. Při každé změně v kódu je ale nutné celý program znovu zkompilovat. Zkompilovaný program jde přímo spustit na systému pro který byl zkompilovaný, jeho spuštění na jiném systému je obtížné, či zcela nemožné. Například program pro počítač se systémem Windows a procesorem Intel nepůjde přímo spustit na počítači se systémem MacOS a procesorem Apple. Kompilované programovací jazyky jsou například C, C++, Rust a Pascal.

Druhou skupinou jsou interpretované jazyky. Zdrojový kód se překládá při

spuštění programu na počítači uživatele, což umožňuje snadnou přenositelnost programu mezi platformami. Program lze spustit na libovolném zařízení, na kterém je nainstalovaný interpret jazyka. Kvůli tomu je ale program obvykle pomalejší. mezi interpretované jazyky se řadí například Java, PHP a Python. [25], [26]

3.4 Výsledek

S ohledem na požadavky na program byl pro tvorbu nového řešení vybrán kompilovaný jazyk. Použití interpretovaného jazyka by sice mohlo usnadnit vývoj aplikace, ale důležitější je vyšší rychlost kompilovaných jazyků. S ohledem na rychlost a využití paměti se jako nejvhodnější kandidáti jeví C, C++ a Rust.

Jazyk C je v porovnání nejrychlejší, jedná se ale o poměrně jednoduchý jazyk který neumožňuje objektově orientované programování. Programování v C je poměrně náročné, v mnoha případech je také kód v C snadno napadnutelný. Program bude pracovat s velkým množstvím dat, které je potřeba ukládat. Objektově orientované programování v takovém případě usnadní vývoj. Proto bylo použití jazyka C zamítnuto.

Jazyk C++ je objektově orientovaný jazyk, oproti C také umožňuje jednodušší zpravu paměti.

Jazyk Rust byl vyvinut s důrazem na bezpečnost kódu s ohledem na použití v kritických aplikacích. Vzhledem k povaze a použití programu není vyšší bezpečnost kódu důležitá.

S ohledem na lepší výkon v binárních stromech a menší využití paměti je pro zbytek práce uvažován jazyk C++ jako nejvhodnější. Ačkoli použití jiného jazyka z výběru by s největší pravděpodobností přineslo analogické výsledky. [7]

Studie proveditelnosti

4.1 Přehled a cíle projektu

Projekt spočívá v návrhu nového řešení výpočtu distribuce tepla v budovách metodou ustálených tepelných toků ve společnosti Rekom, v. d. Klade si za cíl prozkoumat možnost vytvoření a nasazení efektivnější implementace výpočtu. Ta zajistí zrychlení procesu výpočtu a odstranění repetitivních úkonů při výpočtu.

4.2 Analýza a návrh řešení

4.2.1 Analýza požadavků

Cílem analýzy požadavků je vymezení funkčnosti systému, vyjasnění zadání, zachycení omezení a odhad množství práce. Požadavky dělíme na funkční, které postihují chování a vlastní funkcionalitu systému, a nefunkční, které definují vlastnosti systému jako celku a omezení. Do kategorie nefunkčních požadavků tedy patří například požadavky na efektivitu či použitelnost. [14]

4.2.1.1 Funkční požadavky

Funkční požadavky vychází z analýzy zadání společnosti Rekom, v. d.

P1 Formát vstupních a výstupních souborů

Bylo rozhodnuto, že formát vstupních souborů bude kompatibilní se stávajícím řešením. Jako vhodná se jeví konverze tabulky do formátu

csv, ze kterého bude provedeno načtení dat. Výstupní soubor může být v libovolném čitelném formátu.

P2 Kontrola integrity dat

Aplikace musí být schopná kontrolovat úplnost a správnost vstupních dat. V případě nalezení chyb poskytne návodné informace vedoucí ke snadnějšímu nalezení vadných údajů.

4.2.1.2 Nefunkční požadavky

P3 Uživatelská přívětivost

Aplikace musí být jednoduchá na použití po krátkém zaškolení. Měla by urychlovat práci zkušeným uživatelům, například neobsahovat zdlouhavé volení cesty souboru a rutinní varovné hlášky.

P4 Lokální funkčnost v aktuálním prostředí

Program musí fungovat na běžných počítačích s operačním systémem MS Windows 10. Zároveň musí fungovat lokálně, bez nutnosti připojení k internetu či instalaci zvláštního prostředí.

P5 Přenositelnost

Musí být zajištěno snadné použití na různých počítačích se systémem MS Windows 10. Nejlépe pouhým přenesením spustitelného adresáře.

P6 Podpora a udržitelnost

Aplikaci musí být poskytnuta podpora v délce trvání jednoho roku. Zdrojový kód musí být přehledný a dobře zdokumentovaný, po dokončení bude společnosti Rekom, v. d. volně k dispozici k případným úpravám.

4.2.2 FURPS analýza

FURPS je metoda k určení kvality programu. Zkratka FURPS značí functionality (funkčnost), usability (použitelnost), reliability (spolehlivost), performance (výkon) a supportability (udržitelnost). Ve fázi vývoje programu je cílem této analýzy zjistit pokrytí klíčových oblastí existujícími požadavky.

V tabulce 4.1 je vidět, že všechny oblasti metody FURPS jsou pokryty alespoň jedním požadavkem, při sběru požadavků tedy nebyla vynechána žádná důležitá oblast.

Tabulka 4.1: Pokrytí FURPS požadavky

	P1	P2	P3	P4	P5	P6
F	X	X				
U			X	X	X	
R				X		
P			X			
S						X

4.2.3 Návrh řešení a naplnění požadavků

Při zkoumání stávajícího řešení se ukázalo, že úprava, která by odstranila všechny jeho nedostatky, by pravděpodobně znamenala celkovou přeměnu výpočtu s výraznějším použitím VBA. Proto bylo s ohledem na sesbírané požadavky zvoleno vytvoření nového systému v jazyce C++ s využitím matematických podkladů stávajícího řešení.

4.2.3.1 Zvolená technologie

Pro vývoj nového řešení byl zvolen jazyk C++, protože se jedná o univerzální a velmi rychlý jazyk. Programy vytvořené v jazyce C++ se kompilací překládají do strojového kódu, což umožňuje snadnou přenositelnost mezi počítači se stejnou architekturou a systémem i bez instalace doplňků, což je klíčové pro splnění požadavku P4 a P5.

4.2.3.2 Návrh řešení

Aplikace bude obsahovat jednoduché grafické uživatelské rozhraní. To umožní volbu vstupního souboru. Vstupní soubor bude možné alternativně zvolit jeho přetažením do okna programu. Výstupní soubory se zapíší do stejného adresáře, ve kterém se nachází vstupní soubor. Spuštění výpočtu bude možné klávesovou zkratkou, stejně jako otevření výstupního souboru a zavření okna programu po dokončení výpočtu. Rozhraní okna bude navrženo minimalisticky bez zbytečných příkras, dle požadavku P3.

Vstupní data se podle požadavku P1 načtou ze souboru ve formátu csv, který bude možné exportovat přímo z tabulky pro zaznamenávání výsledků měření. Výstupní soubor bude rovněž ve formátu csv a bude vytvořený tak, aby jej bylo možné snadno importovat do programu MS Excel. Program bude v průběhu výpočtu zaznamenávat diagnostické údaje, které budou složité k hledání chyb v datech, což je část požadavku P2. Tyto údaje se uloží do samostatného textového souboru.

Po načtení dat do programu se provede kontrola jejich správnosti. Další kontroly budou probíhat v dílčích krocích výpočtu podle kritérií, která se upřesní před zahájením vývoje. V případě detekce chyby v datech se výpočet přeruší a uživatel bude upozorněn na problém zvýrazněným textem. Součástí chybové hlášky bude stručná informace o chybě. Tato funkčnost naplňuje požadavek P2.

4.2.3.3 Naplnění požadavků

Navržené řešení splňuje všechny funkční i nefunkční požadavky.

4.3 Vývoj a nasazení systému

4.3.1 Zdroje

Společnost Rekom, v. d. nemá kapacity pro samostatný vývoj nového systému. Systém je navržen jako bezúdržbový, vzorce a výpočet zůstávají stejné a navržená technologie umožňuje dlouhodobé používání systému bez dalších úprav. Jako nejvhodnější se tedy jeví vytvoření systému na zakázku, s krátkým obdobím podpory ze strany vývojáře. Výpočet není příliš obtížný z programátorského hlediska, je však vhodné najmout alespoň mírně zkušeného programátora, aby se předešlo zbytečným průtahům a začátečnickým chybám.

4.3.2 Časový plán

Vývoj systému s dodržением všech požadavků, včetně koordinačních schůzek se zástupci společnosti a testováním systému by neměl překročit jeden měsíc, pro účely kalkulace nákladů uvažujeme 20 pracovních dní, každý o 7 odpracovaných hodinách, celkem tedy 140 MH.

Po celou dobu vývoje bude společností Rekom, v. d. zajištěna součinnost v podobě jednoho určeného zaměstnance, který musí mít pravomoc přijímat rozhodnutí týkající se systému. Během prvního týdne poskytne konzultace v rozsahu 20 MH, ve zbylých třech týdnech pak po 5 MH.

Součástí dohody bude podpora systému po dobu trvání jednoho roku. Ta bude zahrnovat bezplatné odstranění nalezených vad. Kromě toho může společnost Rekom, v. d. v průběhu jednoho roku využít až 70 MH pro doplnění dodatečných požadavků.

4.3.3 Finanční a ekonomická analýza

4.3.3.1 Náklady

Tabulka 4.2 obsahuje soupis nákladů související s vývojem a zavedením nové implementace výpočtu od zahájení vývoje po ukončení podpory jeden rok po nasazení. Pro účely kalkulace počítejme s využitím celých 70 MH technické podpory. Neméně důležité je určit ohodnocení vývojáře. Průměrný měsíční plat C++ programátora se blíží k 60 000 Kč, v Praze pak k 70 000 Kč. Pracovní nabídky se obvykle pohybují mezi 40 000 a 100 000 Kč v závislosti na požadované zkušenosti. Protože společnost Rekom, v. d. sídlí v Praze a platy se neustále zvyšují, přikloníme se ve výpočtu k ohodnocení 70 000 Kč za měsíc, což odpovídá 500 Kč / MH. [15],[16]

Tabulka 4.2: Tabulka nákladů

Položka	Množství (MH)	Cena za jednotku	Součet
Vývoj systému	140	500,00 Kč	70 000,00 Kč
Součinnost zaměstnanců	35	700,00 Kč	24 500,00 Kč
Podpora systému	70	500,00 Kč	35 000,00 Kč
Celkem			129 500,00 Kč

4.3.3.2 Výnosy

Přínos nového řešení spočívá především v úspoře času zaměstnanců. V současnosti stráví zaměstnanec v průměru 200 minut, tedy přibližně 3 MH, sledováním a spouštěním jednotlivých kroků výpočtu. Díky kontrole integrity dat se také sníží chybovost výpočtu, eliminuje se manuální procházení dat a obtížné hledání chyb. Nové řešení bude fungovat samostatně, při každém výpočtu tedy uspoří 3 MH času zaměstnanců, kteří budou moci vykonávat jinou činnost. Ve firmě Rekom, v. d. se hodnocením efektivit budov zabývají čtyři zaměstnanci. Každý dokončí v průměru 1,5 zakázky za týden, každý projekt obsahuje jeden výpočet. Celkem tedy provádí v průměru přibližně 300 výpočtů za rok, což odpovídá 900 MH. V tabulce 4.3 je vidět, že úspora dosáhne až 630 000 Kč za rok.

Tabulka 4.3: Tabulka úspor

Položka	Množství (MH)	Cena za jednotku	Součet
Čas zaměstnanců	900	700,00 Kč	630 000,00 Kč
	Celkem		630 000,00 Kč

4.4 Zhodnocení rizik a příležitostí

4.4.1 Rizika

Projekt nové implementace výpočtu si klade za cíl především optimalizovat existující procesy ve společnosti. Pro fungování společnosti nepřináší žádnou výraznou změnu, proto s sebou nese velmi málo rizik. Hlavní riziko pro zhodnocení projektu představuje především nutnost efektivně využít uspořený čas zaměstnanců. Pokud by společnost nebyla schopná tohoto času využít, reálné úspory by byly nižší, než v této studii kalkulované.

4.4.2 Příležitosti

Uvolnění kapacit zaměstnanců umožní společnosti zvýšit objem zakázek i bez najímání dalších pracovníků. Nasazení nového řešení také přinese oprostění zaměstnanců od nepříjemné repetitivní činnosti, čímž dojde k zatraktivnění pracovní pozice.

4.5 Shrnutí výsledků

Navrhované řešení přinese vysoké výnosy již během prvního roku po nasazení. V tabulce 4.4 je ukázáno, že úspora po odečtení nákladů dosáhne přibližně 500 000 Kč již během prvního roku. Rizika nejsou velká a nehrozí prodražení projektu ani další náklady. Realizaci projektu tak lze jednoznačně doporučit.

Tabulka 4.4: Souhrn nákladů a výnosů

Položka	Množství (MH)	Cena za jednotku	Součet
Vývoj a nasazení systému	245	500,00 Kč	-122 500,00 Kč
Výnosy	900	700,00 Kč	630 000,00 Kč
	Celkem		507 500,00 Kč

Aktuální stav a možnosti dalšího rozvoje

5.1 Aktuální stav

Společnost Rekom, v. d. má nyní jasnou představu, jaké výhody může přinést vytvoření nového systému. V současnosti společnost pracuje na dokončení vývoje omezovacích podmínek, které jsou třeba pro automatizaci výpočtu. Vývoj a nasazení nového řešení by mohl začít na začátku roku 2023.

5.2 Možnosti dalšího vývoje

Na práci je možné navázat samotnou implementací navrhovaného řešení. Lze se také zabývat optimalizací výpočtu pro konkrétní systém a platformu.

Závěr

Cílem práce bylo seznámit čtenáře s problematikou distribuce tepla v budovách, prozkoumat současné systémy umožňující efektivně provádět rozsáhlé výpočty a zhodnotit současné řešení. Na základě získaných informací pak zvolit vhodný způsob zefektivnění výpočtů ve společnosti Rekom, v. d. a zpracovat studii proveditelnosti nasazení navrhovaného řešení.

V první kapitole nás práce seznamuje s problematikou určování tepelných bilancí a představuje výpočet metodou ustálených tepelných toků. Ve druhé kapitole uvádí do situace ve společnosti Rekom, v. d. a podoby technologického zázemí. Následně rozebírá stávající řešení, u kterého zkoumá formát vstupu a výstupu a složitost problému. Nakonec identifikuje slabá místa a překážky v dalším rozvoji. Možnost rozvoje stávajícího řešení hodnotí práce jako obtížně proveditelnou, ve třetí kapitole proto porovná vhodné systémy pro vytvoření nového řešení. Jako nejvhodnější určuje jazyk C++.

Ve čtvrté kapitole je zpracována studie proveditelnosti vytvoření a nasazení navrženého řešení. Její součástí je analýza požadavků, na základě které je navrženo základní fungování programu. Finanční analýza a analýza zdrojů ukázala, že vývoj systému bude trvat 1 měsíc a včetně souvisejících nákladů vyjde přibližně na 130 000 Kč. Systém slibuje ušetřit až 630 000 Kč za rok, tedy bude vydělávat již během prvního roku po nasazení. Analýza rizik a příležitostí odhalila pouze málo rizik s nízkým dopadem a několik významných příležitostí, projekt je tedy doporučen k realizaci.

Literatura

- [1] REKOM, V. D. *Ústřední vytápění komplexní řešení pro bytové domy*. [online]. [cit. 18.4.2022]. Dostupné z: <https://rekomvd.cz/>
- [2] BLAŽÍČEK, J. *Rozhovor o fungování výpočtu a společnosti*. Nymburk, 31.3.2022
- [3] BLAŽÍČEK, J. *Upřesnění rozhovoru o fungování výpočtu a společnosti*. Nymburk, 2.5.2022
- [4] MERRIAM-WEBSTER. *Man-hour Definition & Meaning*. Dictionary by Merriam-Webster [online]. Merriam. 2022. [cit. 14.4.2022]. Dostupné z: <https://www.merriam-webster.com/dictionary/man-hour> (vlastní překlad)
- [5] MAREŠ, M. & VALLA, T. *Průvodce labyrintem algoritmů*. CZ.NIC, z.s.p.o. Praha 2017. ISBN 978-80-88168-19-5
- [6] PEREIRA, R & COUTO, M & RIBEIRO, F & RUA, R & CUNHA, J & FERNANDES, J. P. & SARAIVA, J. *Energy efficiency across programming languages: how do energy, time, and memory relate?*. ACM Digital Library [online]. Association for Computing Machinery. New York 2017. 256–267. [cit. 20.4.2022]. DOI: <https://doi.org/10.1145/3136014.3136031> (vlastní překlad)
- [7] GEEKFORGEEKS. *Top 10 Fastest Programming Languages*. GeeksforGeeks [online]. [cit. 14.4.2022]. Dostupné z:

- <https://www.geeksforgeeks.org/top-10-fastest-programming-languages/>
(vlastní překlad)
- [8] KOPECKÝ, P. *Materiál a konstrukce. Šíření tepla, vzduchu a vlhkosti v budovách a stavebních prvcích*. [online]. Fakulta Stavební, ČVUT v Praze. Praha 2014. [cit. 18.4.2022]. Dostupné z: http://tzb2.fsv.cvut.cz/vyucujici/16/oppa/124mtib_kopecky.pdf
- [9] BRUNS, D. *23 things you should know about VLOOKUP*. Exceljet [online]. [cit. 20.4.2022]. Dostupné z: <https://exceljet.net/things-you-should-know-about-vlookup> (vlastní překlad)
- [10] KENTON, W. *Visual Basic for Applications (VBA)*. Investopedia [online]. Dotdash Meredith. 2020. [cit. 20.4.2022]. Dostupné z: <https://www.investopedia.com/terms/v/visual-basic-for-applications-vba.asp> (vlastní překlad)
- [11] MICROSOFT CORPORATION. *Získání nápovědy k používání programu Visual Basic Editor*. Microsoft support [online]. [cit. 20.4.2022]. Dostupné z: <https://support.microsoft.com/cs-cz/office/ziskani-napovedy-k-pouzivani-programu-visual-basic-editor-61404b99-84af-4aa3-b1ca-465bc4f45432> (vlastní překlad)
- [12] MICROSOFT CORPORATION. *Getting started with vba in office*. Microsoft support [online]. 22.1.2022. [cit. 20.4.2022]. Dostupné z: <https://docs.microsoft.com/en-us/office/vba/library-reference/concepts/getting-started-with-vba-in-office> (vlastní překlad)
- [13] VITOSH ACADEMY. *VBA vs C++ function - which one is faster in Excel?*. Vitosh Academy. 2017. [cit. 20.4.2022]. Dostupné z: <https://www.vitoshacademy.com/vba-vs-c-function-which-one-is-faster-in-excel/> (vlastní překlad)
- [14] ŽOLTÁ, L. *Disciplína sběr a analýza požadavků*. lucka.zolta.cz [online]. [cit. 20.4.2022]. Dostupné z: <http://lucie.zolta.cz/index.php/software-inzenyrstvi/150-disciplina-sber-a-analyza-pozadavku>
- [15] PROFESIA. *Nabídky práce: Programátor C++* profesia.cz [online]. [cit. 2.5.2022]. Dostupné z: <https://www.profesia.cz/prace/programator-cpp/> (vlastní překlad)

- [16] INDEED. *Kolik si vydělá Developer C/C++ v Česko?*. cz.indeed.com [online]. 28.4.2022. [cit. 2.5.2022]. Dostupné z: <https://cz.indeed.com/career/c++-developer/salaries> (vlastní překlad)
- [17] BLAŽÍČEK, J. *Trochu teorie neuškodí*. Polohové koeficienty pro spravedlivé rozúčtování nákladů [online]. © 2019 - 2022. [cit. 5.5.2022]. Dostupné z: <https://www.peklo.net/teplo-v-bytovych-domech/>
- [18] BLAŽÍČEK, J. *Polohové koeficienty*. Polohové koeficienty pro spravedlivé rozúčtování nákladů [online]. © 2019 - 2022. [cit. 5.5.2022]. Dostupné z: <https://www.peklo.net/polohove-koeficienty/>
- [19] BLAŽÍČEK, J. *Spotřeba tepla a její měření*. Polohové koeficienty pro spravedlivé rozúčtování nákladů [online]. © 2019 - 2022. [cit. 5.5.2022]. Dostupné z: <https://www.peklo.net/spotreba-tepla/>
- [20] SVĚT OKEN s. r. o. *Co to jsou tepelné ztráty a jak se počítají*. svetoken.cz [online]. 17.10.2019. [cit. 7.5.2022]. Dostupné z: <https://www.svetoken.cz/plastova-okna/co-to-jsou-tepelne-ztraty-a-jak-se-pocitaji>
- [21] AMBROŽOVÁ, I & HORÁK, P. *Stanovení tepelných zisků zasklení ze slunečního záření v energetickém hodnocení budov*. tzb-info.cz [online]. 27.8.2012. [cit. 7.5.2022]. Dostupné z: <https://www.tzb-info.cz/energeticka-narocnost-budov/8972-stanoveni-tepelnych-zisku-zaskleni-ze-slunecniho-zareni-v-energetickem-hodnoceni-budov>
- [22] MACHINE DESIGN. *What's the Difference Between Conduction, Convection, and Radiation?*. machinedesign.com [online]. 31.8.2015. [cit. 7.5.2022]. Dostupné z: <https://www.machinedesign.com/learning-resources/whats-the-difference-between/document/21834474/whats-the-difference-between-conduction-convection-and-radiation> (vlastní překlad)
- [23] KRÁLOVÁ, M. *Vedení - kondukce tepla*. Techmania Science Center Eduporál [online]. Plzeň 2007. [cit. 7.5.2022]. Dostupné z: <http://edu.techmania.cz/cs/encyklopedie/fyzika/struktura-latek/vnitri-energie/vedeni-kondukce-tepla>
- [24] KAŠPAR, J. *Polohové koeficienty a jejich využití*. problemysvj.cz [online]. 12.1.2018. [cit. 7.5.2022]. Dostupné z: <https://www.problemysvj.cz/polohove-koeficienty-pouziti/>

- [25] IBM CORPORATION. *Application programming on z/OS*. ibm.com [online]. © 1990 - 2010. [cit. 9.5.2022]. Dostupné z: <https://www.ibm.com/docs/en/zos-basic-skills?topic=zos-compiled-versus-interpreted-languages> (vlastní překlad)
- [26] FREECODECAMP. *Interpreted vs Compiled Programming Languages: What's the Difference?*. freecodecamp.org [online]. 10.1.2020. [cit. 9.5.2022]. Dostupné z: <https://www.freecodecamp.org/news/compiled-versus-interpreted-languages/> (vlastní překlad)

Seznam použitých zkratk

MH man-hour, česky člověkohodina, jednotka množství práce, kterou vykoná jeden pracovník za 1 hodinu. [4]

MS Microsoft

GB Gigabajt

RAM Random access memory - operační paměť

VBA Visual Basic for Applications

Obsah přiloženého CD

	readme.txt	stručný popis obsahu CD
	src	zdrojová forma práce ve formátu \LaTeX
	text	text práce ve formátu PDF