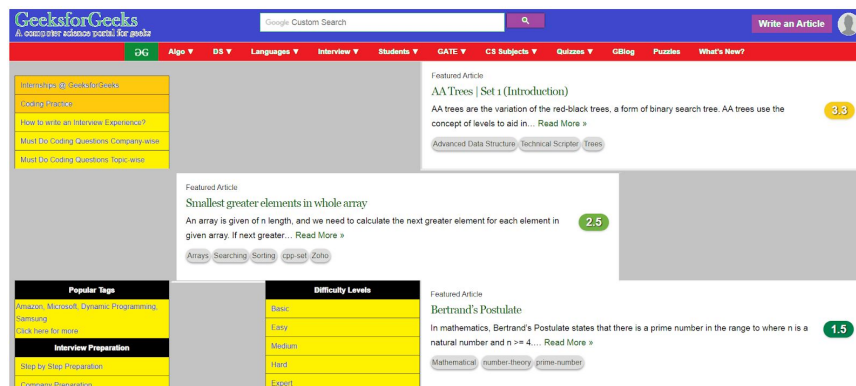


Uživatelská rozhraní cvičení 1

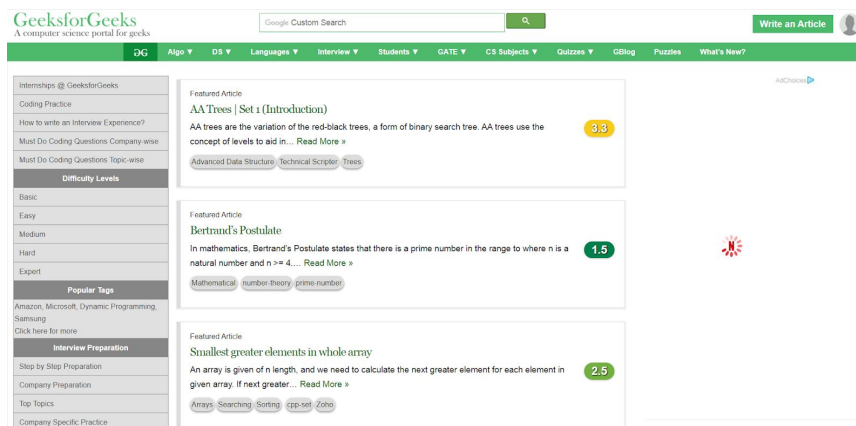
Základní informace

Cílem kurzu je seznámit studenty se zásadami tvorby uživatelských rozhraní. Můžete mít nápad na skvělou aplikaci, můžete ji mít špičkově naprogramovanou funkčnost, ale pokud nebude mít uživatelsky přívětivé grafické rozhraní, nebude úspěšná.

Když se např. podíváte na následující dvojici obrázků, tak se asi shodneme, že s aplikací, případně webovou stránkou, která je na spodním obrázku by se nám pracovalo lépe. Narozdíl od horního rozhraní má vhodně zvolené odstíny barev (není dobré kombinovat barvy různých odstínů), má jasně danou nabídku v levé části obrazovky a jednotlivé prvky jsou zarovnané a nepůsobí tak rušivě jako v případě horního rozhraní.



BAD UI



GOOD UI



zdroj: <https://www.geeksforgeeks.org/user-interface-ui/>

Další ukázkou špatného a správného uživatelského rozhraní lze vidět na následujícím obrázku, kde se jedná o mobilní aplikaci. Levému rozhraní opět chybí zarovnání prvků, také volba

návratu na předchozí stránku je vyřešena elegantněji pomocí šipky v pravém rozhraní než pomocí těžko rozpoznatelného tlačítka v případě levého rozhraní.



zdroj: <https://www.ascentspark.com/ui-ux>

Pravidla, jak správně navrhnout uživatelské rozhraní a čemu se vyhýbat, se dozvíte na přednáškách. Ve cvičení tohoto předmětu si vyzkoušíte tvorbu uživatelských rozhraní pomocí knihoven pro tvorbu GUI v jazycích Python (tkinter) a C++ (Qt). Tyto knihovny nabízí grafické prvky (popisky, tlačítka, textová pole, zaškrťovací políčka, apod.). Vaším úkolem na cvičeních bude vytvořit grafické rozhraní pro různé aplikace s využitím těchto prvků a za dodržení pravidel představených na přednáškách.

Požadavky a bodování

- Dodělení příkladů ze cvičení (0-15 bodů)
předpokladem je 8 lehkých příkladů – 4 v Pythonu, 4 v C++
- Odevzdání projektu (Python) (16-30 bodů)
- Odevzdání projektu (C++) (16-30 bodů)
- Závěrečný test (10-25 bodů)

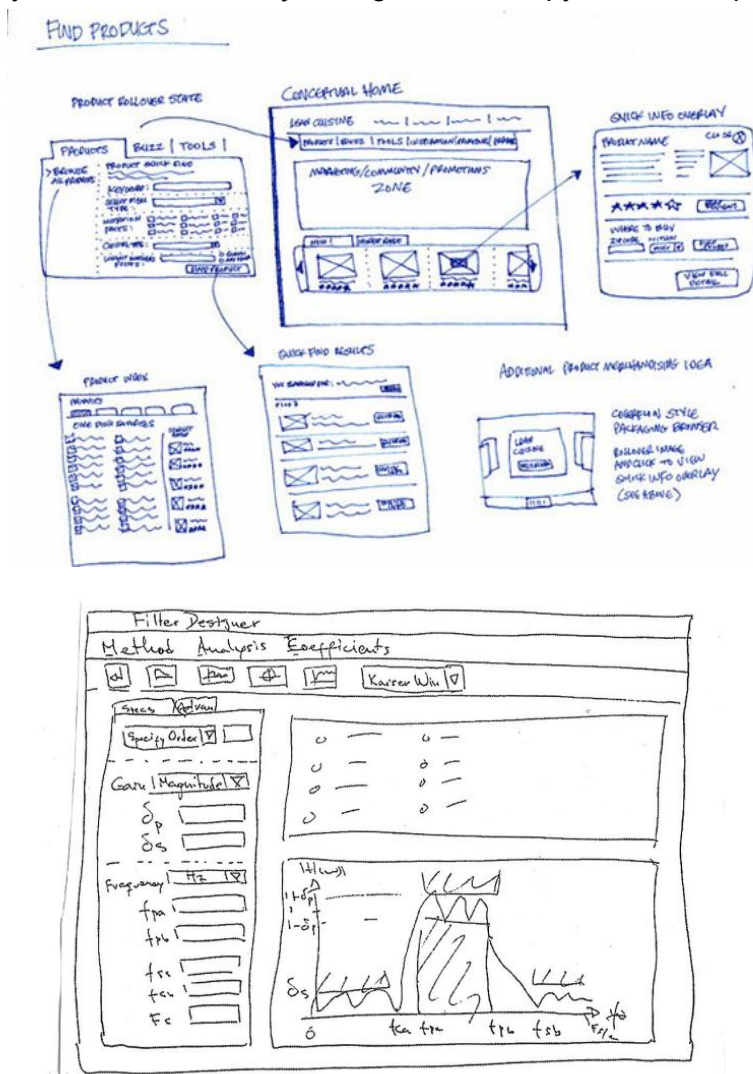
Odevzdávání projektů

Projekty se budou odevzdávat přes webové rozhraní kelvin.cs.vsb.cz.

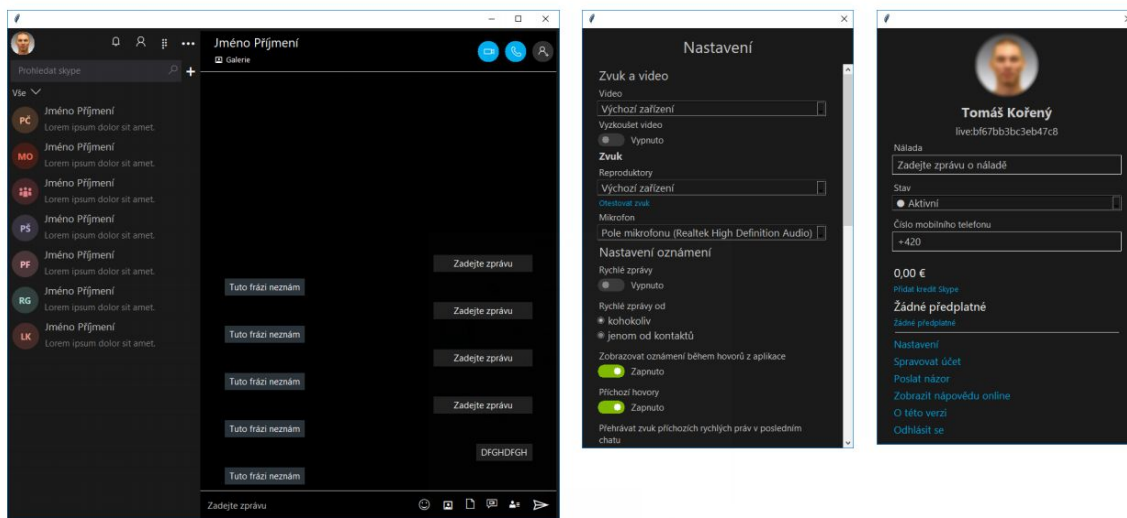
Projekt

Cílem projektů je ukázat, že dovedete vytvořit „pěkně“ vypadající uživatelské rozhraní podle pravidel probraných na přednáškách. Projekt by měl demonstrovat uživatelské rozhraní vámi vybraného programu – buď existujícího nebo klidně vlastní aplikace (program nemusí být plně funkční, jedná se hlavně o GUI). Projekt by měl obsahovat minimálně tři rozsáhlejší okna.

Nejprve vytvoříme návrh na papíře (či v nějakém grafickém programu) – u návrhu je třeba promyslet, jak prvky v okně rozvrhnout, jak reagovat na vstupy uživatele, apod.



Ve výsledku pak může projekt vypadat třeba takhle (autor student Tomáš Kořený, 2018):



Předběžný harmonogram projektu:

Cvičení 3: Návrh projektu

Cvičení 7: Odevzdání projektu v jazyce Python

Cvičení 13: Odevzdání projektu v jazyce C++

V obou jazycích budete pracovat na stejném projektu.

Úkol na cvičení 1: seznámení se s knihovnou tkinter v jazyce Python

Jazyk Python je multiplatformní (Unix, Windows, Mac OS) interpretovaný objektově orientovaný programovací jazyk. S tímto jazykem se budete seznamovat v předmětu Skriptovací jazyky, nicméně u tvorby GUI si postačíme se základy, proto nezapomínejte, že tento jazyk zatím neznáte.

Knihovna tkinter obsahuje základní prvky pro budování grafického uživatelského rozhraní (GUI).

tkinter: <https://docs.python.org/3/library/tkinter.html>

tkinter.ttk: <https://docs.python.org/3/library/tkinter.ttk.html> – novější knihovna umožňující snadnější stylování prvků a obsahující některé prvky navíc (např. Notebook, TreeView)

Dokumentace např. zde: <https://anzelg.github.io/rin2/book2/2405/docs/tkinter/index.html>

Pro pochopení práce s knihovnou jsou připraveny jednoduché ukázky na některé základní prvky. Ty jsou ke stažení zde: http://mrl.cs.vsb.cz/data/vyuka/URO/tk_examples.zip

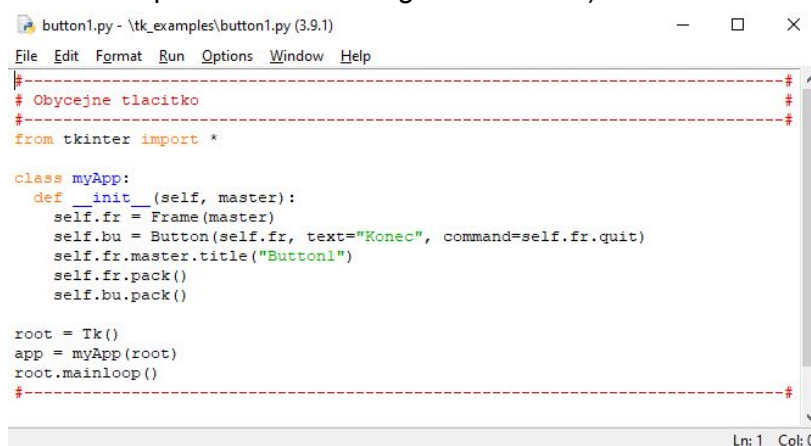
Uložte si ukázky na disk, vyzkoušejte si je, zkuste některé i upravit.

Instalace a spuštění na OS Windows:

Nainstalujte si Python (www.python.org), kde aktuální verze je 3.9.1.

Stáhněte si ukázkové příklady a zkuste si je spustit.

Zdrojové *.py soubory můžete editovat ve vhodném textovém editoru nebo přímo v IDLE (Python's Integrated Development and Learning Environment).



```
button1.py - \tk_examples\button1.py (3.9.1)
File Edit Format Run Options Window Help
#-----#
# Obycejne tlacitko
#-----#
from tkinter import *

class myApp:
    def __init__(self, master):
        self.fr = Frame(master)
        self.bu = Button(self.fr, text="Konec", command=self.fr.quit)
        self.fr.master.title("Button1")
        self.fr.pack()
        self.bu.pack()

root = Tk()
app = myApp(root)
root.mainloop()
#-----#
Ln: 1 Col: 0
```

Instalace a spuštění na OS Linux (Debian, Ubuntu, Mint):

sudo apt install python3-tk // instalace tk pro python3

cd tk_examples/ // přepnutí se do adresáře se skripty

python3 soubor.py // spuštění skriptu

Např. skript *button1no.py* vykreslí tlačítko. Zdrojový kód je minimalistický, bez použití třídy v Pythonu a obsahuje pouze tyto řádky:

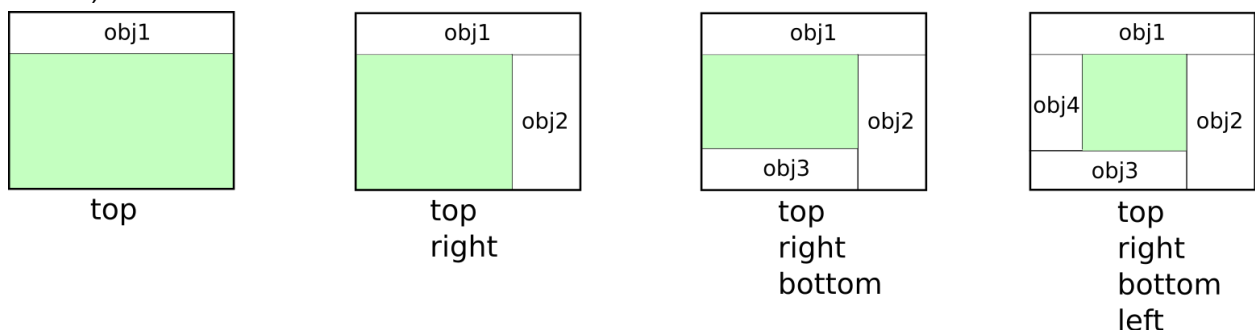
```
-----  
from tkinter import *    # import knihovny tkinter  
  
top = Tk()               # vytvoří prázdné okno a přiřadí do proměnné top  
b = Button(top, text="Konec", command=top.quit) # vytvoří tlačítko v rámci okna top, s textem Konec a  
s příkazem pro ukončení po kliknutí na toto tlačítko - v tuto chvíli se ale ještě nevykreslí, pouze se uloží  
do proměnné b  
top.title("Button 1")    # nastaví název okna  
b.pack()                 # vykreslí tlačítko do okna (této funkci se budeme věnovat více v dalším cvičení)  
  
top.mainloop()          # smyčka pro nekonečné vykreslování okna  
-----
```

Zkuste do okna přidat druhé tlačítko, tzn. vytvořte nový objekt třídy **Button** a ten pomocí funkce **pack** vykreslete. Kde se tlačítko vykreslí? Tlačítka budou pod sebou. To lze změnit a to pomocí parametru **side** ve funkci **pack**. Defaultně je tento parametr nastaven na **top**, tzn. že se tlačítko (obecně jakýkoliv objekt nebo kontejner) vykreslí do horní části plochy volné ke kreslení. Plocha volná pro kreslení se mění podle toho, jak se objekty přidávají. Pokud tedy vykreslíme nějaký prvek pomocí **obj.pack(side=top)**, další prvek už nelze umístit nad **obj**. Parametr **side** lze nastavit na hodnoty **top**, **bottom**, **left**, **right**. Pořadí vykreslování ovlivní jak prostor pro dané objekty, tak volný prostor pro další prvky. K pochopení poslouží následující příklad.

Mějme 4 grafické prvky, které vykreslíme v následujícím pořadí:

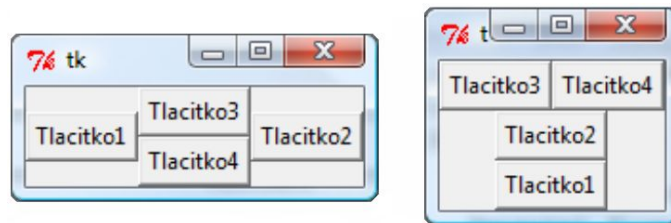
```
obj1.pack(side=top)  
obj2.pack(side=right)  
obj3.pack(side=bottom)  
obj4.pack(side=left)
```

Rozložení prvků a volného prostoru pro další prvky se pak mění následovně (volná plocha je zeleně).



Pokud přidáme páté tlačítko, tak kam se vloží?

Zkuste prvky rozvrhnout tak, aby odpovídaly rozložení v obrázku.



Na příštím cvičení se podíváme i na další parametry funkce pack, které ovlivní vzhled a chování vykreslovaných prvků nebo kontejnerů.